CHARACTERIZATION AND AVOIDANCE OF IN-FIELD DEFECTS IN SOLID-STATE IMAGE SENSORS

by

Jozsef Dudas
B.A.Sc. Simon Fraser University 2005

THESIS SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF APPLIED SCIENCE

in

The School
of
Engineering Science

Spring 2008

APPROVAL

Name: **Jozsef Dudas**

Degree: **Master of Applied Science**

Title of Thesis: **Characterization and Avoidance of In-Field Defects in Solid-State Image Sensors**

Examining Committee:

Chair: **Dr. Albert Leung**
Professor

____

Dr. Glenn H. Chapman
Senior Supervisor
Professor

____

Dr. Ash M. Parameswaran
Supervisor
Professor

____

Dr. Marinko V. Sarunic
Internal Examiner
Assistant Professor, Engineering Science

Date Defended: March 31st, 2008

ABSTRACT

As solid-state image sensors become ubiquitous in sensing, control and photography products, their long-term reliability becomes paramount. This thesis experimentally examines the nature of in-field faults and demonstrates two combined hardware-software approaches for detecting and mitigating them. Characterization experiments found that most tested commercial cameras developed hot pixels that create image bright spots and degrade dynamic range. Faults appear spatially point-like and uniformly distributed, and they develop continually over time. Silicon displacement damage, induced by terrestrial cosmic rays, is the likely cause.

A fault tolerant active pixel sensor is developed to isolate hot defects to a portion of the pixel, enabling software algorithms to correct the faults without sacrificing dynamic range. Experimentally-emulated hot pixels can be corrected within ±5% error.

A new statistical software approach is developed to identify and calibrate stuck and abnormal-sensitivity faults from only regular photographs. Monte Carlo simulations verify the detection accuracy in complex environments.

Keywords: image sensor; fault tolerance; hot pixel; active pixel sensor; reliability

Subject Terms: Digital cameras; Imaging systems; Semiconductors--Failures; Semiconductors--Reliability; Integrated circuits--Fault tolerance

ACKNOWLEDGEMENTS

I would like to thank my supervisor, Dr. Glenn Chapman, for giving me the opportunity to explore so many new ideas and projects. Your steady guidance and endless patience throughout many challenges are much appreciated. I would also like to extend my gratitude to Dr. Israel Koren and Dr. Zahava Koren for their insight and guidance in all things fault tolerant, particularly in relation to the on-line fault identification work. I owe many thanks to my examining committee, Dr. Ash Parameswaran and Dr. Marinko Sarunic, especially for the humour they always offered during my time at SFU.

This work could not have been completed without those who laid the foundations for my research and others that offered assistance along the way. To Cory, Michelle, Jenny and the rest of the sardine can crew: Thank you.

Finally, I am grateful to my parents, friends and Josna for your unwavering support throughout my studies. I could not have completed this journey without you!

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

4RR / 8RR	4 / 8 Round-Robin
APS	Active Pixel Sensor
BCCD	Buried-Channel CCD
CCD	Charge-Coupled Device
CDS	Correlated Double Sampling
CIS	CMOS Image Sensor
CMOS	Complimentary Metal Oxide Semiconductor
CTE	Charge Transfer Efficiency
DSC	Digital Still Camera
DSLR	Digital Single Lens Reflex
FFCCD	Full Frame CCD
FPN	Fixed Pattern Noise
FTAPS	Fault Tolerant Active Pixel Sensor
HCCD	Horizontal CCD
IS-ITCCD	Interlace Scan Interline Transfer CCD
ITCCD	Interline Transfer CCD
MP	Megapixel
PD	Photodiode
PPD	Pinned Photodiode
PRNU	Photo Response Non-uniformity
PS-ITCCD	Progressive Scan Interline Transfer CCD
RTS	Random Telegraph Signal
VCCD	Vertical CCD

CHAPTER 1

INTRODUCTION

Digital imaging has rapidly transformed how we interact with the physical world around us. With applications extending across arts and entertainment, safety, and security, the possibilities offered by electronic vision seem endless, and the demand for associated products only continues to grow. Digital photography is a prime example. Standing in any museum, concert recital, or park with throngs of people snapping countless photographs leaves no doubt that digital still cameras (DSCs) have been embraced by consumers. The Camera and Imaging Products Associate (CIPA) representing the major manufacturers in the industry reports that more than 42 million DSCs were shipped in the first half of 2007 alone[1] with annual sales projected to climb to 103 million units by 2009[2].

Seemingly less widespread than photo cameras, recently developed automotive vision systems remain hidden from users while potentially preventing accidents. Lane Departure Warning (LDW) systems, for example, record video of the road ahead of the vehicle and warn the driver when he is about to drift unexpectedly from the lane. With the success that LDW systems have already experienced in commercial trucks, passenger vehicles manufacturers, like Nissan Motors and General Motors, have recently begun incorporating LDW in their own vehicles[3-5]. Other products in varied stages of development include simple back-up assistance video, pedestrian detection, and driver drowsiness detection[6].

Further behind the scenes, automation and inspection systems help protect the integrity of the goods we buy. In carmakers' factories, robots detect components in three dimensions in order to expertly manoeuvre these components into place[7]. Large-scale bakeries analyze photos of bread products to ensure a perfectly toasted crust[8]. Even agricultural distributors scan potatoes and other products to guarantee that only high quality vegetables are delivered to stores[9].

While these systems appear to be wildly different, at the heart of each one is a solid-state image sensor that delivers visible-light image data. Continued development of the sensor technology, in parallel with signal processing algorithms and hardware, has been critical in enabling the end products. Both dominant imager technologies, Charge Coupled Device (CCD) and CMOS Image Sensor (CIS) arrays, have evolved to meet a diverse set of requirements.

The majority of electronic imaging applications extract fine details from high-quality source images. Therefore, most research to date has focused on bringing optical performance in line with that of traditional photographic film cameras. In the field of digital photography, this research has led to sensors with vastly improved spatial resolution, usable light sensitivity that exceeds film, and excellent uniformity across the entire image area. Automotive sensors, meanwhile, must deliver equally usable images whether operating in direct sunlight, the black of night, or a combination of both, such as when a car passes through tunnels. This requirement has led to the development of very wide-dynamic range imagers that have optimum performance in all lighting environments[10].

In other products, factors beyond pure image quality must be considered. The need for small and inexpensive embedded cameras in mobile phones has been one part of the driving force behind highly-integrated CIS arrays[11].

All of these advances have contributed to the ubiquitous nature of electronic imaging. However, as sensor technologies mature and are deployed in an increasingly large number of products, reliability must become a primary concern. All microelectronics are known to degrade over time and imagers are no exception. As a great deal of study has been devoted to failures in standard digital and analog circuits, many of the supporting function blocks in an imager chip are already well understood. However, the optoelectronic nature of sensor pixels places them in a separate category that has received comparatively little attention in the literature. Change in a pixel's characteristics over the time that it is deployed in the field can lead to an unacceptable reduction in image quality. Coupled with the fact image sensors are often larger than even microprocessors, making them more likely to develop defects, this sensitivity to even small variations in performance makes defects a serious issue. In addition, image sensors are analog devices so changes that would not affect classic digital devices (e.g. small threshold shifts in transistors, etc) can have a significant impact.

Concern for the lifetime of cameras may seem superfluous when many electronics are conveniently discarded and upgraded. Nonetheless, consider the example of automobile lane-departure warning systems, where some manufacturers must plan on the cars having a lifetime well in excess of a decade. The safety components of the vehicle, camera systems included, should guarantee reliable operation for the same period as the automaker will be liable for degradation-based failures. Similarly, plant automation equipment usually involves large capital investment, which carries with it the expectation of long product lifetimes.

Even in the DSC market, camera lifetime is expected to become a more prominent factor in purchasing decisions. Until recently, rapid growth in DSC sales could be attributed to users frequently upgrading their existing cameras to those with improved performance (usually spatial resolution, i.e. "more Megapixels"). However, modest improvements to recently-released new models indicate the pace of future progress will slow down, and frequent replacement may become less attractive to consumers. Moreover, the CIPA reports show than an increasingly larger proportion of sales are being captured by higher performance and higher priced Digital Single Lens Reflex (DSLR) cameras[2]. We can conclude that consumers are becoming more interested in making a long-term investment in photographic equipment, and they will correspondingly expect the quality of their equipment to match that investment. Therefore, the longevity of digital cameras will have to match that expectation.

The objective of this thesis is to address the void in reliability information available for commercially off-the-shelf image sensors such as those deployed in ordinary digital still cameras. This goal is achieved in three complementary steps by:

- Quantifying the characteristics of performance-degrading defects in real sensors;
- Experimentally demonstrating a pixel design for mitigating the most common type of failure;
- Developing a novel algorithm for detecting and studying defects in existing sensors.

A brief introduction to the motivation and goals of each point is given in the section below.

1.1.1 Defective Pixel Characterization

The development of faulty pixels is considered the primary failure mode of image sensors, particularly when those faults arise after a camera leaves the manufacturing plant. A key contribution of this thesis is in determining whether commercial DSCs are likely to suffer faulty pixels throughout their lifetime and in quantifying the extent of that degradation. Until now, much of the attention on this topic has focused on imagers operating in harsh environments like space (see [12-14] for examples), while commercial DSC data has been limited to anecdotal reports of defective pixels from photographers. This shortcoming is addressed in Chapter 3 with an analysis of numerous commercial DSCs operating in common, benign terrestrial environments. Using standard calibration techniques, both the quantity and behaviour of faulty pixels are characterized. These results give a first estimate of the prevalence of defective pixels, and the accompanying analysis provides insight into possible mechanisms that cause faulty pixels, allowing more robust sensor designs to be developed.

1.1.2 Fault Tolerant Pixel

The Fault Tolerant Active Pixel Sensor (FTAPS) was proposed by Chapman and Audet[15] as a means of improving the manufacturing yield of image sensors. Work on the FTAPS has since focused on evaluating its imaging performance. Sensors have been tested without faults[16-19] and with simple fully-stuck faults[20-23]. However, Chapter 3 shows that more complex "hot pixels" are the dominant failure in commercial cameras, and Chapter 4 shows how the FTAPS can be applied to mitigate these faults as well. The concept is experimentally evaluated via two methods: by electrically simulating "hot pixels" in a custom designed FTAPS array implemented in 0.18 μm CMOS technology; and by optically simulating hot pixels in a standard FTAPS array previously implemented in 0.35 μm CMOS technology. A recovery algorithm is also described and tested.

1.1.3 In-field Fault Identification Algorithm

Many failure modes beyond the dominant hot pixel type are conceivable in all imagers, with each fault further reducing image quality. Such failures may be corrected by software if the faults can be correctly located and identified, but fault identification through sensor calibration is a complex procedure that is costly and impractical to implement in the field. To add robustness to existing ordinary camera systems, Chapter 5 describes a novel algorithm that can automatically identify these faults on-line, in the field, using only images captured by the cameras, without any special sensor or calibration hardware. The algorithm operates by using Bayesian inferences to match pixel values from images to those expected by mathematical defect models. The resulting probabilities are accumulated over a moderate sequence of images to reduce false positive tests experienced by competing algorithms. Monte-Carlo simulations demonstrate the effectiveness of this technique in mitigating a range of fault types.

1.2 Summary

Solid-state image sensors form the core electronic imaging systems that have enabled so many creative applications. While the optical performance of imagers has been the primary focus of imager research until now, camera technology is reaching maturity, thereby placing an increasing importance on sensor reliability. This thesis aims to provide quantitative data on performance degradation in commercial sensors and to provide a solution to combat defects in future design iterations.

The remainder of this thesis is organized as follows: Chapter 2 provides background on the physical operation of modern commercial image sensors. In addition, pixel defects and possible mechanisms for the development of defects are introduced in this chapter. Previous work on fault tolerant active pixel sensors is described here as well. Chapter 3 describes experiments that characterize in-field defects found in common DSCs and describes various analysis techniques used to propose which mechanisms are responsible. Chapter 4 describes how the Fault Tolerant APS can mitigate hot pixels and presents experimental verification for this idea. Finally, Chapter 5 describes and demonstrates a novel algorithm that automatically identifies a range of defect types simply by analyzing standard images captured on a DSC. Concluding thoughts and directions for future work are offered in Chapter 6.

CHAPTER 2

SOLID-STATE IMAGE SENSOR BACKGROUND

In order to model and evaluate how imager defects influence performance, we need to understand the fundamental principles behind solid-state image sensors. This section describes the basic operation of commercially available variants from two imaging technologies: Charge Coupled Device (CCD) and CMOS Image Sensor (CIS) arrays. We begin by describing the light detection mechanism shared by both technologies before providing details of their unique readout mechanisms. These operating principles are then used to build a simple model of image sensor response that can be used to extract the most crucial metrics of imager performance. In light of these models, we examine possible defect mechanisms and their influence on pixel behaviour. Finally, a fault tolerance mechanism previously proposed for simple defect types is introduced.

2.1 Light Detection

The photoelectric effect forms the basis of light detection in any semiconductor sensor. When a photon flux impinges on a semiconductor, some of the light may be absorbed if the photon energies are larger than the material bandgap, as given by

$$E_{photon} = h\nu \geq E_g, \tag{2.1}$$

where E_{photon} is the photon energy, h is Planck's constant, v is the photon frequency, and E_g is the bandgap of the semiconductor. The absorbed energy from a photon elevates an electron from the valence band to the conduction band, thereby creating free carriers. The generation process is illustrated in Figure 2.1 for an ideal semiconductor with valence band, conduction band and forbidden gap energies E_v, E_c, and E_g respectively. One result of this relationship is that material properties place an upper bound on the wavelength of light that can be absorbed. Silicon, for example, has a bandgap of approximately 1.1 eV so that it is sensitive up to wavelengths of about 1100 μm, corresponding to the near-infrared range.

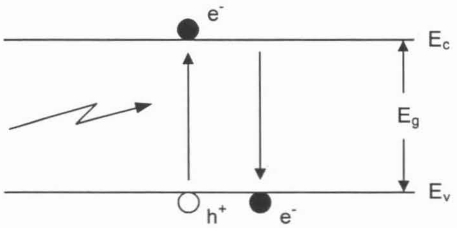


Figure 2.1. Photo generation of carriers in an ideal semiconductor.

This absorption is a statistical process that continues through the bulk of the material such that the remaining photon flux, $\Phi(z)$, at a given depth, z, is governed by the exponential Beer-Lambert relationship

$$\Phi(z) = \Phi_0 \exp(-z\alpha), \tag{2.2}$$

where Φ_o is the incident flux at the surface and α is a wavelength-dependant absorption coefficient (measured in cm^{-1}) that is unique to each material. Light at shorter wavelengths has a larger absorption coefficient owing to its higher energy and is thus more likely to be fully absorbed at a shallow depth. Conversely, long wavelength red light has a small absorption coefficient, allowing it to penetrate deep into the semiconductor and requiring a thick material to absorb the light completely. Table 2.2 gives some typical values for the absorption coefficient, α, as well as the absorption length, α^{-1}, which shows the depth at which 63% of the incident light energy has been absorbed[24]. We can readily see that the majority of illumination is absorbed within several micrometers of the silicon surface.

Table 2.1. Absorption of visible light in Silicon[24].

Colour	*Wavelength* *(nm)*	*Absorption coefficient* $\alpha\ (cm^{-1})$	*Absorption length* $\alpha^{-1}\ (\mu m)$
Red	600	3.75×10^3	2.67
Green	525	7.07×10^3	1.41
Blue	450	1.98×10^4	0.51

However, simply generating free carriers is insufficient to measure light because several carrier recombination mechanisms occur in parallel with photogeneration, meaning that charges disappear before they can be measured. For example, the average minority carrier lifetime is less than 1 μs in a modern silicon substrate. To counter this recombination, carriers can be captured in the electric field of a depleted region of semiconductor.

2.1.1 Photodiodes

Most modern imagers utilize the depletion region of a reverse-biased photodiode[25], pictured in Figure 2.2, for charge accumulation.

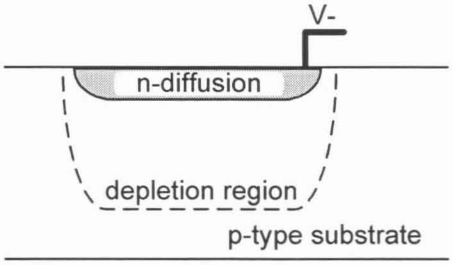


Figure 2.2. Structure of a photodiode and process of charge collection.

Device operation begins in a reset phase in which a positive voltage is applied to V_{PD}, thereby maximizing the diode depletion region. Next, the reset bias is removed and the diode is allowed to float. Light penetrating to the depth of the depletion region creates free electron-hole pairs. The intrinsic electric field sweeps these carriers out of the depletion region, separating them to opposite ends of the photodiode, thereby inducing an instantaneous photocurrent. In addition, free carriers generated outside the depletion region but sufficiently nearby (i.e. within several diffusion lengths as appropriate for the material and carrier type) may diffuse toward the depletion region and contribute to the photocurrent.

However, the photocurrent is usually on the order of picoamperes and is difficult to measure directly. Instead, the process is allowed to continue for some finite duration, called the *integration time*, while charge is integrated (collected) in the diode, as described by the following expression:

$$Q(T_{\text{int}}) \approx Q_{reset} - \left(I_{photo} \cdot T_{\text{int}}\right), \tag{2.3}$$

where T_{int} is the integration period, $Q(T_{int})$ is the charge remaining on the photodiode after integration, Q_{reset} is charge on the diode immediately after reset, and I_{photo} is the induced photocurrent that is assumed to remain constant. Note that the above result is only a first-order approximation. In reality, I_{photo} varies throughout the integration period due to the change in volume of the photodiode collection region (i.e. the reverse-biased diode becomes discharged) so that fewer optical carriers are captured, especially from deep-penetrating long-wavelength (redish) light.

This collection process can continue until the diode is completely discharged (i.e. integrated optical charge exceeds Q_{reset}). Therefore, very low intensity illumination can be measured by collecting light over longer periods. Finally, the magnitude of collected charge is read by the rest of the imaging system and the collection process begins again at the reset phase.

Nonetheless, non-optical generation mechanisms put an upper bound on the maximum integration time. Even in an ideal semiconductor, spontaneous thermal generation creates free carriers that are readily captured by the photodiode, resulting in *dark current*. Therefore, the diode is discharged even in the dark and its capacity to collect further optically generated carriers is reduced. In real devices, defects in the silicon lattice give rise to “trap” energy states in the forbidden band, which enable trap-assisted thermal generation to proceed more rapidly. The result is a "hot pixel fault" discussed in a later section. Equation (2.3) for the charge collected during integration can thus be updated to include a term for dark current, I_{dark}, that includes all dark generation sources:

$$Q(T_{\text{int}}) = Q_{reset} - T_{\text{int}}\left(I_{photo} + I_{dark}\right). \tag{2.4}$$

While bulk silicon tends to be highly regular, the fabrication process can create defects at the interface between different materials, like at the silicon surface. Therefore, modern sensors move the charge-collection region of the photodiode away from the surface by implementing the pinned photodiode (PPD) structure[26] shown in Figure 2.3. Here, an extra p+ layer is implanted at the surface to shield the charge collection volume from carriers generated at the surface. To reset the PPD, a transfer gate, *TX* in Figure 2.3, completely moves all charge from the collection volume to a floating diffusion (FD) region. Note that the PPD operates entirely in the charge domain, such that all of the collected carriers are transferred to the floating diffusion (where they are converted to a measurable voltage), regardless of the PPD capacitance.

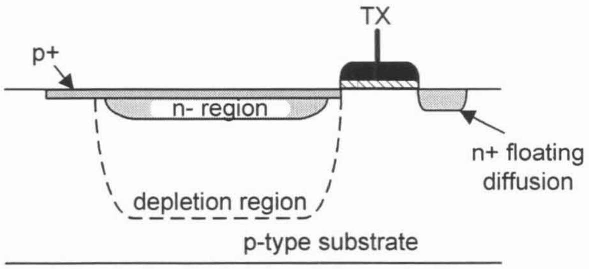
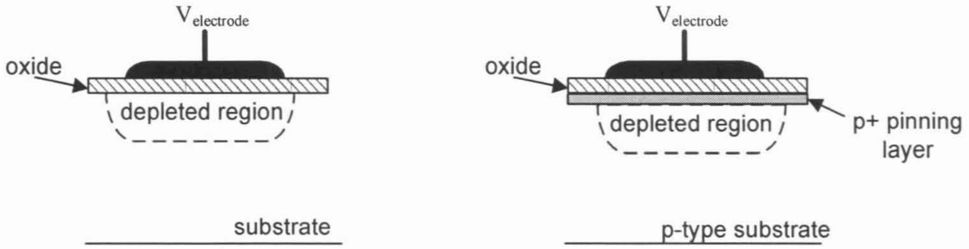


Figure 2.3. Simplified structure of a pinned-photodiode.

2.1.2 Photogates

A photogate is a metal oxide semiconductor (MOS) capacitor used to collect photo carriers. Photogates are light-sensitive CCDs but have also been implemented in CIS systems. To create the charge collection region, a positive voltage is applied to the electrode of the structure in Figure 2.4(a), which depletes the lightly doped substrate and creates a potential well under the gate, thereby allowing photogenerated carriers to be collected. The gate electrode material is typically made transparent to maximize light sensitivity.

Much like the photodiode, defects at the substrate-oxide interface can lead to enhanced dark current, so many sensor designs utilized the buried MOS diode structure pictured in Figure 2.4(b). The extra implant shifts charge collection deeper into the substrate and away from dark current sources.



(a) Standard Photogate.

(b) Buried Photogate.

Figure 2.4. Simplified structure of (a) a photogate and (b) a buried photogate.

The photogate is also a purely charge domain device with a linear illumination response, such that the total charge accumulated during integration is given by the familiar expression:

$$Q(T_{\text{int}}) = T_{\text{int}}\left(I_{photo} + I_{dark}\right). \tag{2.5}$$

Once we have captured light and stored it as charge, it needs to be communicated to the rest of the imaging system. A useful image sensor will include a matrix of many photosites, and each one must be read to form an image. This readout methodology is the primary difference between the two dominant imaging technologies that are described in the following two sections.

2.2 CMOS Image Sensors

CMOS Image Sensors (CIS), previously also referred to as CMOS Active Pixel Sensors (APS), are a relatively new imaging technology that encompass a wide range of architectures.

They were first described by Noble in 1968[27] but their development remained largely stagnant until the mid-1990's when reduced CMOS device geometries and improved fabrication technology made the sensors viable[28]. Modern CIS are created in a standard or slightly modified CMOS fabrication process that gives them several advantages over competing methods:

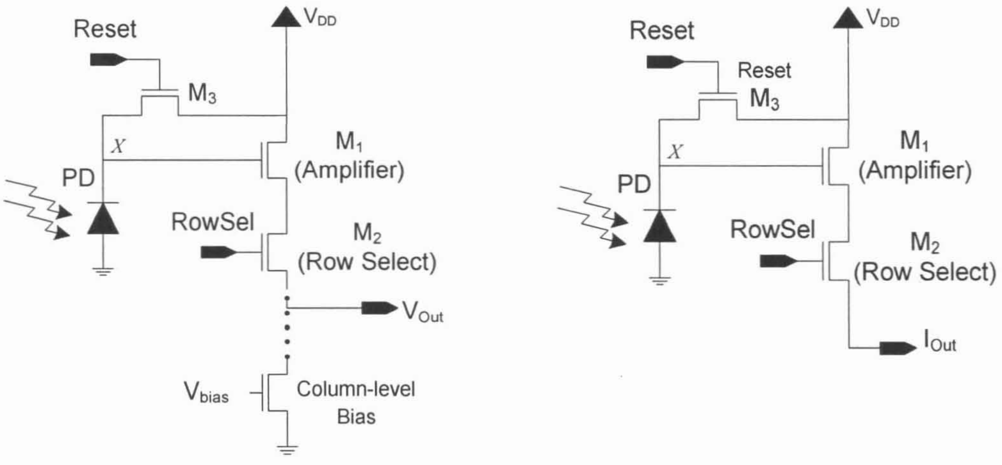
- Low production costs through leveraging existing, mature CMOS foundries;
- Straightforward integration of control, signal processing and support electronics;
- Low power consumption due to reduced power supplies and simplified control signalling (especially over CCDs).

These benefits, and others, can be realized to varying degrees depending on the application and its requirements.

CIS have led to many creative designs that maximize parameters like intra-scene dynamic range and imaging speed. However, the architectures that find widespread use in DSCs are the simple variants of the Active Pixel Sensor (APS): the 3-transistor (3T APS) and 4-transistor (4T APS) devices.

2.2.1 Three-Transistor APS

The three-transistor APS is the classic design and is the most straightforward. Figure 2.5 shows schematic diagrams of the 3T APS configured for both voltage and current domain operation. Both designs utilize a simple photodiode, *PD*, a single transistor amplifier, M_1, a row-select transistor, M_2, and a reset transistor, M_3. In the preferred voltage domain operation, the amplifier is configured as a source follower such that V_{out} closely follows the photodiode voltage, *V(X)*, with good linearity, nearly unity gain and a small offset. When M_1 is configured as a transconductance amplifier, the 3T APS operates in the current domain with the drawback that the output current, I_{out}, is a non-linear function of the photodiode terminal voltage. This is the operating mode employed in the fault tolerant APS described in Section 2.5.5.



(a) Voltage domain. *(b) Current domain.*

Figure 2.5. Schematic diagram of 3T APS in (a) voltage-domain and (b) current-domain.

The control and output signal waveforms shown in Figure 2.6 provide an overview of the image capture process at a single pixel, which is comprised of the following steps:

1. Reset photodiode *PD* through reset transistor M_3 at time t_{reset}.
2. Collect photogenerated carriers for T_{int} seconds (the integration period) on PD and get of amplifier M_2.
3. Transfer output signal to column line by activating row-select transistor, M_2.
4. Read and store output signal.
5. Reset *PD* again.
6. Read and store reset signal.
7. Subtract measurements from 4 and 6 to calculate pixel value.

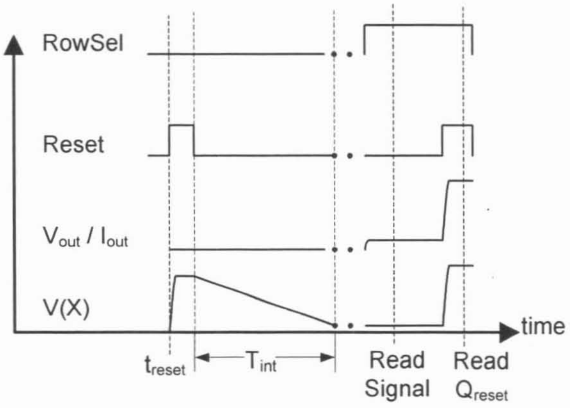


Figure 2.6. Control signal waveforms for a 3T photodiode APS.

In more detail, imaging begins by resetting the photodiode at time t_{reset}, when transistor M_3 is pulsed on to charge *PD* to the power supply level, V_{DD}. During the following integration interval, which lasts for a period of T_{int}, light striking *PD* is converted to charge according to Eq. (2.3). This charge is effectively collected on the combined capacitances at node *X* in Figure 2.5, causing a voltage to develop at *X*:

$$V(X) = \frac{Q_{reset} - T_{\text{int}}\left(I_{photo} + I_{dark}\right)}{C_X}, \tag{2.6}$$

where C_X is the combined total capacitance at *X* due to the photodiode depletion and junction capacitances, all of the gate capacitances of M_1, and any other parasitic capacitances due to wiring. In practice, the photodiode capacitance is about 10-times greater than the other parasitic components of C_X.

After integration, each pixel in a column is accessed individually by activating the row select transistor, M_2. In a voltage domain device, immediately after integration, the source follower output follows the expression

$$V_{out} = a\frac{Q_{reset} - T_{\text{int}}\left(I_{photo} + I_{dark}\right)}{C_X} + \Delta_{SF}, \tag{2.7}$$

where *a* is the source follower gain, which is typically slightly less than unity, and Δ_{SF} is the small offset of the same device. This value for V_{out} is stored by column-level sample-and-hold circuitry. The pixel is then reset again and the output value stored to estimate Q_{reset}. Finally, the difference is taken between the reset value and the signal value to remove any fixed offsets that arise due to variations across in-pixel devices. This process is called *double sampling*. Consequently, the recorded pixel output value is a linear function of illumination, dark current, integration time, and some noise components that are beyond the scope of this discussion:

$$V_{pixel} = a\frac{T_{\text{int}}\left(I_{photo} + I_{dark}\right)}{C_X} + < Noise >. \tag{2.8}$$

Additional components of the imaging system may then apply further processing like noise suppression, analog gain, and analog-to-digital conversion. Operation of the current domain 3T APS is similar with the exception of the non-linear response of the in-pixel amplifier, which results in a less intuitive result.

2.2.2 Four-Transistor APS

As was noted, the Pinned Photodiode (PPD) is currently the preferred detector technology because of its superior dark performance. Because of its charge-domain operation, the PPD requires a 4T APS architecture, a schematic diagram of which is shown in Figure 2.7 for voltage domain operation. This architecture again includes a source-follower amplifier, a row-select transistor for random access readout of pixels, and a reset transistor. However, unlike the 3T APS, the potential of charge integrated on the PPD cannot be read directly. Here, a transfer gate, M_4, is used to move charge from *PPD* to a floating diffusion segment, *FD*, which causes a potential to develop at the gate of the amplifier transistor.

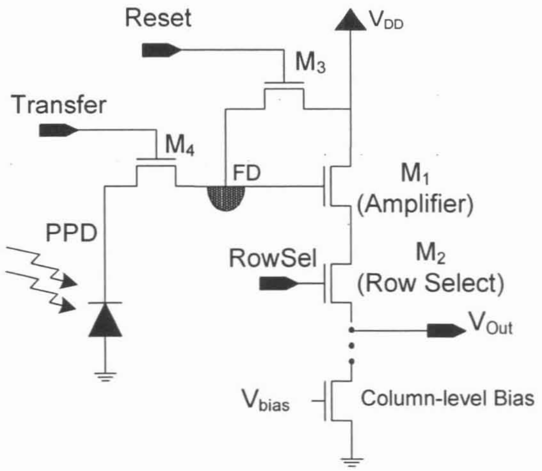


Figure 2.7. Schematic diagram of a 4T photodiode APS.

Control signal timing and the resulting outputs for a single, simplified capture cycle of the 4T APS are shown in Figure 2.8. The complete capture process can be summarized as follows:

1. At time t_{reset}, reset photodiode *PD* by transferring all charge through transfer gate M_4.
2. Collect photogenerated carriers for T_{int} seconds (the integration period) on PD.
3. Reset floating diffusion through reset transistor M_3.
4. Record reset charge, Q_{reset}.
5. Transfer signal charge to floating diffusion.
6. Read and record signal charge.
7. Subtract values from 4 and 6 to calculate final pixel value (Correlated Double Sampling).

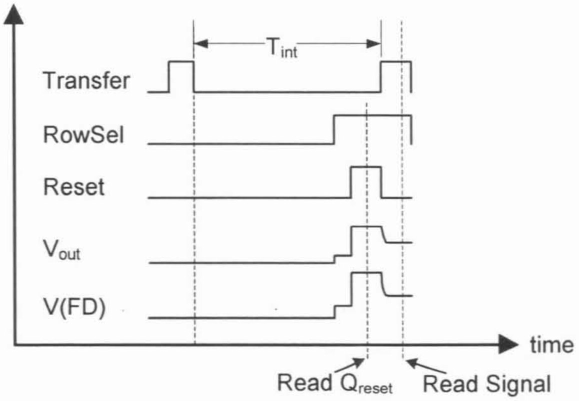


Figure 2.8. Control signal waveforms for 4T photodiode APS.

In more detail, operation begins by resetting *PPD* by pulsing *Transfer* to remove all charge in the diode. Because of the fixed pinning potential of *PPD,* charge transfer is complete[26], thus resetting *PPD* at the same time. During integration, the transfer gate, M_4, is turned off and charge is collected in *PPD*. Readout of a single pixel begins by pre-charging *FD* through M_3. *RowSel* is also activated and the reset value is buffered by the source follower (described in Section 2.2.1) before being stored by sample and hold circuitry. Finally, all collected charge from *PPD* is transferred to *FD*, creating a potential given by,

$$V(FD) = \frac{Q_{reset} - T_{\text{int}}\left(I_{photo} + I_{dark}\right)}{C_{FD}}, \tag{2.9}$$

where C_{FD} is the combined capacitance of the floating diffusion, the amplifier gate, and parasitic elements; the remaining terms of the above equation were given in Eq. (2.6). This signal voltage is also sampled by the sample and hold circuit and subtracted from the reset value in a process called *Correlated Double Sampling* (CDS). In this case, the sampling is correlated because the signal is read immediately after the reset value so the temporal noise component of Q_{reset} is common to both values. The final stored pixel output is again a linear function of illumination, dark current and integration time as given earlier in Eq. (2.8) because the 4T APS operates in fundamentally the same manner as the 3T APS.

2.2.3 CIS Arrays

Both of the CMOS APS pixel architectures are typically implemented as a matrix of pixels like the system shown in Figure 2.9, albeit varying levels of integration. Pixels can be randomly accessed because each one in includes a photosensing element, charge-to-voltage conversion element, and an amplifier (i.e. the simple source-follower). Typically, the imager is arranged in a column-parallel architecture, such that output signals from a column of pixels are multiplexed together and fed to a dedicated sample and hold circuit for use in the CDS process. Some designs may also incorporate on-chip analog-to-digital converters, including dedicated column-parallel devices, although the cameras tested in Chapter 3 integrate only the analog components prior to digital conversion. Discussion of the exact circuit used for each block varies significantly between sensors but additional information can be found in [29].

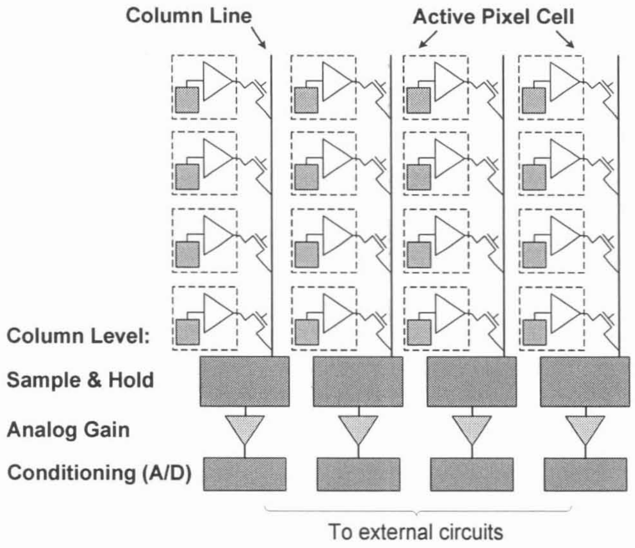


Figure 2.9. Block diagram of a typical CMOS APS array.

2.3 Charge-Coupled Device Arrays

Despite the many apparent advantages of CIS, Charge Coupled Devices (CCD) continue to be widely employed due to their maturity and reliable performance. CCDs were first conceived and implemented by Boyle and Smith at Bell Labs in 1969 for use as analog memories, but they quickly discovered that CCDs could be charged by visible light via the photoelectric effect and were thus useful as solid-state imagers[30]. Since their inception, CCD sensor designs have evolved significantly, but current commercial chips primarily use the multiphase clocking variation of the Interline Transfer CCD (ITCCD) or Full Frame CCD (FFCCD) designs[31]. Many manufacturers do not reveal which particular design is implemented in their chips, but the ITCCD and FFCCD dominate the high-end market because of their superior performance.

Both systems split imaging into three distinct functions. Light collection takes place in either a photodiode or photogate, charge transfer occurs in the Vertical CCD (VCCD), and charge readout is done by the Horizontal CCD (HCCD). Further system processing tasks like analog gain and digital conversion are typically performed off-chip and will not be discussed here. Because the design and operation of photosensors were described Section 2.1, the remainder of this section will focus on the basics of CCD operation.

2.3.1 Charge Transfer

The CCD is primarily used to move charge around the semiconductor. Its basic structure is an array of closely-space MOS capacitors with overlapping dual gates to allow careful control of the semiconductor surface potential everywhere. Individually, each CCD cell is the same as the photogate described in Section.2.1.2. A unit cell for a simple 4-phase CCD is shown in Figure 2.10; this same structure is repeated many times to build a VCCD or HCCD.

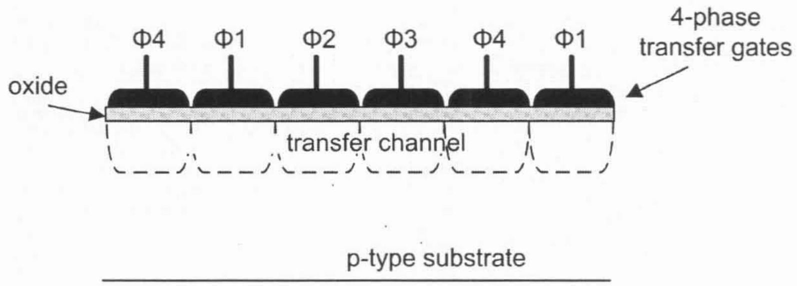
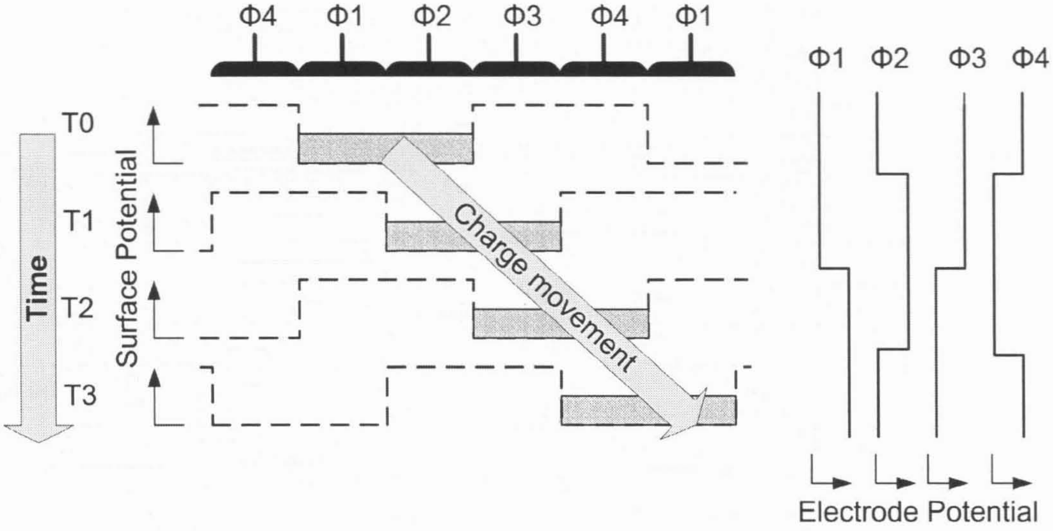


Figure 2.10. Physical structure of a basic 4-phase charge-coupled device (CCD).

To move charge, the voltage applied to each electrode is varied in order to raise or lower the surface potential under the electrodes. Diffusion and drift transport mechanisms then cause charge to move from positions of high potential to low potential. Figure 2.11 demonstrates the process of moving a packet of charge along the VCCD; Figure 2.11(a) shows the surface potential underneath each electrode at each instant in time, while Figure 2.11(b) illustrates the voltage applied to each electrode phase.



(a) Surface potentials. *(b) Electrode potentials.*

Figure 2.11. Charge transfer process in a CCD as it proceeds over time. (a) Surface potential underneath each transfer electrode. (b) Voltage applied to each electrode, after Yamada[31].

Initially, at time *T0*, a high voltage is applied to electrodes Φ2 and Φ3, while a low voltage is applied to Φ1 and Φ4, creating a potential well under the centre of the CCD. In this state, charge can be transferred to the resultant well from an attached photodiode, but the charge cannot be transferred laterally to other pixels. To transfer charge along to the next pixel, the electrodes are toggled according to the pattern of Figure 2.11 until the charge packet reaches its next destination.

Note that deep trap states in the CCD channel can inhibit charge movement between wells, affecting the charge transfer efficiency (CTE), which quantifies the percentage of charge that is successfully transferred. This limitation is particularly important when considering that charge must be shifted across thousands of wells in a readout operation. Even if 99.9% CTE is achieved in individual cells, only 35.9% of the charge remains after shifting across 1024 CCD phases, which corresponds to a relatively small sensor array. As discussed earlier, Si-SiO$_2$ interface traps are a major source of traps in the channel, thus most commercial sensors modify the CCD structure to use a buried CCD (BCCD) akin to the structure described earlier in Figure 2.4(b). Defects in the silicon bulk within the CCD well also reduce CTE but with lessened impact because such damage is typically not repeated across an entire row of pixels.

2.3.2 Interline Transfer CCD

We can now put together a simplified interline transfer CCD (ITCCD) sensor shown in Figure 2.12. The device utilizes a matrix of pinned photodiodes to capture light and accumulate charge. Arrays of 4-phase BCCDs are placed in vertical stripes between columns of photodiodes and are hence called the vertical CCDs (VCCD). A single row of CCDs, called the horizontal CCD (HCCD), is placed at the end of the VCCDs and is used to move charge packets toward the output amplifier. Both the HCCD and VCCDs are shielded from light by metal conductor layers so that the photodiodes are the only photosensitive elements in the system.

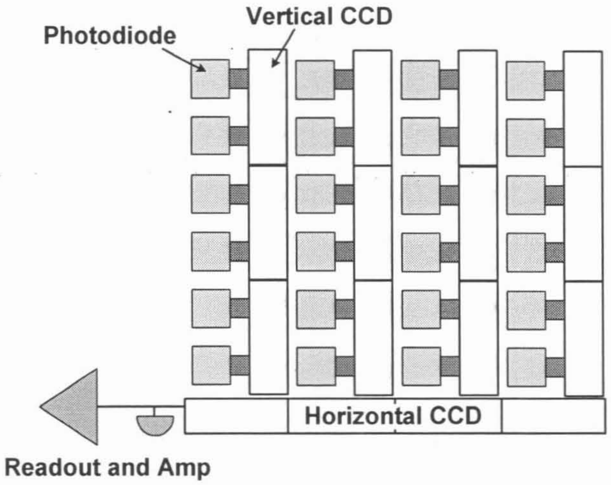


Figure 2.12. Block diagram of an industry-standard interline transfer CCD image sensor, after Yamada[31].

Operation begins by transferring all charge out of the PPDs, then light integration proceeds as described earlier. Immediately after this period, charge is transferred from the photodiodes to the optically-shielded VCCDs so that an electronic shuttering effect is achieved. Next, the

VCCDs shift charge down by one row so that the bottom-most row is transferred into the HCCD, which in-turn rapidly transfers packets into the output amplifier. This process repeats until each row of charge is shifted into the HCCD.

Pixel readout may be performed in one of two sequences, depending on the complexity of the sensors: Progressive-Scan ITCCDs (PS-ITCCD) transfer all of the pixels at once, as described above; conversely, simpler Interlace-Scan ITCCD (IS-ITCCD) arrays first transfer charge from photodiodes in even-numbered rows and then transfer charge from the odd rows of the sensor.

Regardless of the scan sequence, charge-to-voltage conversion is done pixel-serially at the end of the HCCD by a single combination of a floating diffusion and a source-follower amplifier (sometimes multiple amplifiers are placed at each end of the CCD). Therefore, despite the complex circuits involved, the stored pixel output is again a linear function of illumination intensity, integration time and dark current, as given by the now-familiar equation:

$$V_{pixel} = a\frac{T_{\text{int}}\left(I_{photo} + I_{dark}\right)}{C_{FD}}, \tag{2.10}$$

where a includes the gain of the source-follower as well as losses due to less-than-ideal CTE, and C_{FD} includes the floating diffusion capacitance and connected parasitic elements.

2.3.3 Full Frame CCD

The basic FFCCD is illustrated in the block diagram of Figure 2.13. Here, the VCCD plays the dual role of combined photosensing element and vertical transfer element, while the rest of the system is the same as the ITCCD. During integration, the VCCDs act like photogates, collecting charge in a potential well. During readout, they behave like shift-register CCDs used in the ITCCD. This simpler structure allows more of the pixel area to be made light sensitive, which improves the overall illumination sensitivity of the imager.

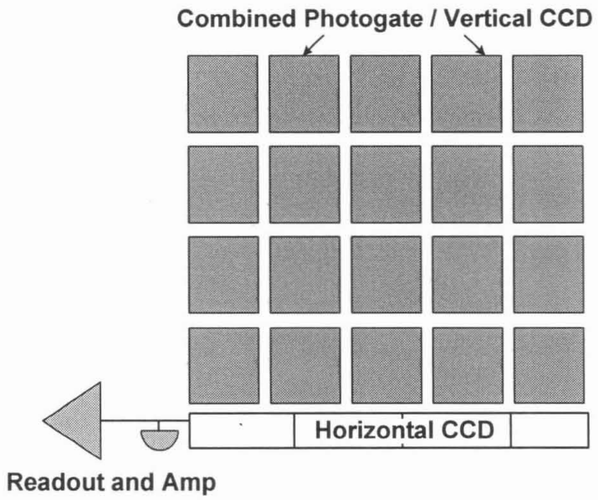


Figure 2.13. Block diagram of a full frame CCD system, after Yamada[31]

2.4 Pixel Response

The preceding discussion showed that, as a first approximation, the imagers of interest respond to light in almost identical fashion. Therefore, we can build the following common model of the desired pixel response:

$$Y = \left\{ \begin{matrix} m \cdot T_{\text{int}} \cdot \left(I_{photo} + I_{dark} \right) + \Delta + < Noise >, & 0 < Y < Y_{Sat} \\ Y_{Sat}, & otherwise \end{matrix} \right\}, \qquad (2.11)$$

where Y is the normalized pixel output (i.e. independent of the physical quantity measured); m is a combined gain term, called the *sensitivity*, that incorporates all appropriate factors, such as collection efficiency of a given wavelength of light, charge-conversion gain in the photodiode or floating diffusion, source-follower attenuation, and any analog gain; Δ is any residual offset not removed by double-sampling; *<Noise>* represents all the collective temporal noise sources in the signal path; and the remaining terms are familiar from previous sections. Y_{Sat} indicates the saturated output level (a value of 1 in this normalized system) that is reached when the potential well of the photo-collection element (i.e. photodiode or photogate) has been completely filled with electrons. Once the pixel reaches this state, the output remains constant and further incident illumination cannot be measured.

Note that analog gain, and consequently the sensitivity, m, is programmable in most imagers systems to accommodate a variety of scene brightness levels. In photographic cameras, this gain is adjusted in increments such that the camera's absolute light sensitivity conforms to an international standard, International Standards Organization (ISO) 12232:2006, that relates scene luminance to pixel output in several ways. Thus, the programmable gain level is often referred to as "ISO sensitivity" or "ISO speed" with appropriate values.

The relationship in Eq. (2.11) is expressed graphically in Figure 2.14 as an illumination response curve, where the illumination intensity (directly related to I_{photo}) is taken as the independent variable. The curve is given for a fixed T_{int} so that offsets due to dark current and other elements are lumped together.

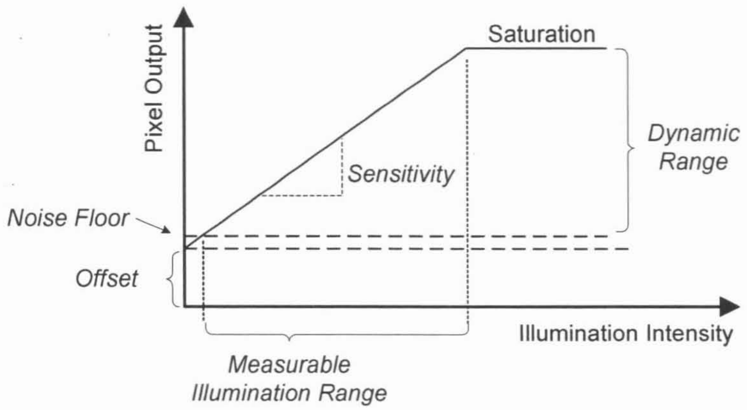
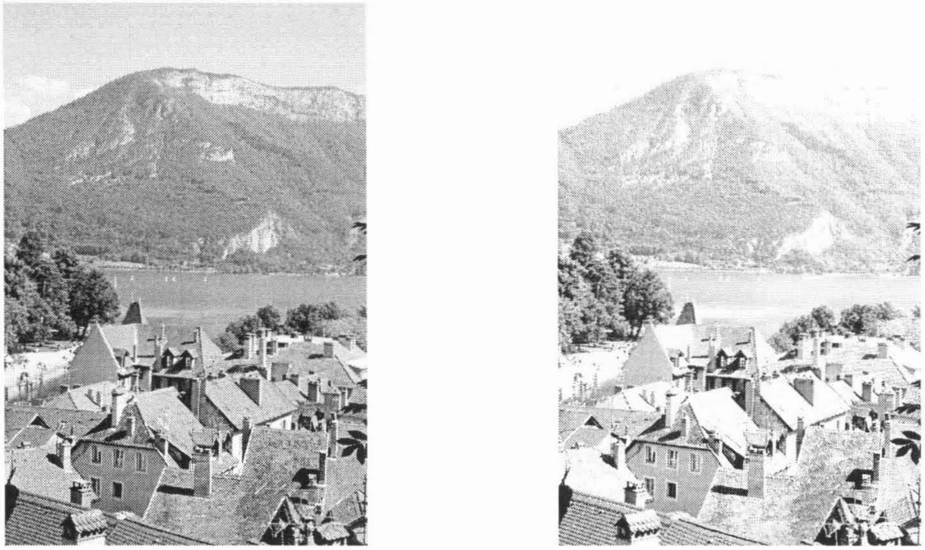


Figure 2.14. Representative illumination response curve of an image sensor.

The key performance measures that can be extracted from Figure 2.14 are sensitivity and dynamic range (DR). Sensitivity is the slope of the response curve, measured as pixel output per illumination intensity.

Dynamic range is the range of usable signal swing at the pixel output and directly determines the range of illumination intensity that can be imaged in a single exposure. Dynamic range is bounded on the lower end by offset and noise floor and on the higher end by the saturation level, which is determined by the pixel capacity, Q_{reset}. Factors that increase offset, like excessive dark current, will reduce dynamic range and consequently the measurable illumination range. Figure 2.15 shows a typical scene imaged with a wide dynamic range sensor and a reduced dynamic range sensor. Clearly, the image with larger dynamic range conveys more detail and demonstrates the importance of reducing dark current-induced pixel offset to maintain dynamic range.



(a) Wide dynamic range image *(b) Reduced dynamic range image.*

Figure 2.15. Scene imaged with (a) wide and (b) reduced dynamic range.

Note that device properties can fluctuate across the area of an image sensor, leading to inter-pixel variations in sensitivity and offset that are collectively called Fixed Pattern Noise (FPN). Specifically, differences in *m* are labelled photo-response non-uniformity (PRNU) while shifts in Δ are termed offset FPN.

2.5 Faulty Pixels

Like all microelectronic devices, image sensors are susceptible to defects in the materials, which can cause undesired pixel behaviour. Damage to key structures like interconnects, gate oxides, or silicon bulk can cause outright pixel failure (e.g. disconnected or short-circuited power supplies) or less obvious but equally severe effects like reduced sensitivity. Circuit failures can be classified as either "hard" or "soft" depending on whether they have a permanent or transient effect. This work considers only hard fails, which are permanent damage to the pixel and which degrade every image captured after they develop. Moreover, although failures can conceivably occur in any functional block of an imager, this thesis considers only effects on the pixels because the photosensing region of the sensor occupies the vast majority of the chip area, making it the most susceptible to failure.

2.5.1 Fault Types

In standard terminology, physical damage to semiconductor structures, such as the displacement of atoms in the silicon bulk, is referred to as a *defect*. The physical process causing the defect is called the *defect mechanism*. The observed erroneous behaviour is termed a *fault*. Here, a faulty pixel is empirically defined as one that either behaves significantly different from Eq. (2.11) or has parameters that deviate from the rest of the sensor.

With reference again to Eq. (2.11), a number of conceivable fault types arise due to large shifts in sensitivity, m, offset, Δ, dark current, I_{dark}, or combinations of these three variables. Table 2.2 gives a summary of the fault types considered in this work. Similar faults can be grouped into families, as listed in the first column of Table 2.2. The second column gives the common name of each fault type, and the *parameter change* column describes which variables in Eq. (2.11) deviate from the norm to cause the given fault. The final column in Table 2.2 describes how that faulty pixel would distort a monochrome (grey scale) image. In order to maintain generality across all sensor technologies, this fault model is completely descriptive and does not specify which defects cause each fault. The impact of these faults on a sample image is shown in Figure 2.16.

Table 2.2. Summary of faulty pixel types.

Fault family	*Fault name*	*Parameter change*	*Description*
Fully stuck	Stuck low	$m = 0$	Always dark
	Stuck high	$\Delta = 1.0, m = 0$	Always bright
	Stuck mid	$0 < \Delta < 1.0$, m = 0	Always grey.
Partially stuck	Partially stuck	$0 < \Delta < 1.0$	Responds to light but never goes black.
Abnormal sensitivity	High sensitivity	$m >> 1.0$	Always brighter than neighbours.
	Low sensitivity	$0 < m << 1.0$	Always darker than neighbours.
Hot	Hot pixel	$I_{dark} >> 0$	Bright spots in long exposures.

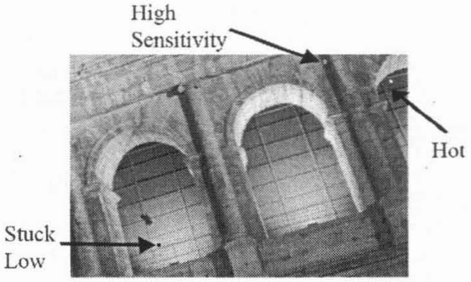


Figure 2.16. Sample scene distorted by standard faulty pixels.

Note that we can classify the above failures as *catastrophic faults* (usually shortened to *faults*) or *parametric faults*, although the distinction is somewhat fuzzy when applied to imaging pixels. Here, catastrophic faults are those whereby the pixel no longer produces any usable output, such as fully stuck pixels. Parametric faults refer to pixels that produce some illumination dependant output with a non-catastrophic shift in some imaging parameter, such as partially stuck, abnormal sensitivity, and hot pixels that have shifts in m, Δ, and I_{dark}, respectively.

The impact of the fully stuck faults (stuck high, low and mid) is the most severe because those pixels cannot measure light and can only create FPN. Similarly, while partially-stuck and hot pixel faults remain sensitive to light, they contribute to increased FPN and a much reduced dynamic range due to the offset component, as explained in Section 2.4. Abnormal sensitivity faults contribute to PRNU.

2.5.2 Defect Mechanisms

Defects can be broken down into two categories based on when they occur: at manufacture time (i.e. before the product is shipped) or after the product is in the field. Manufacture-time defects have been a focus of the semiconductor industry since its inception in order to maximize the yield of usable chips on a wafer and lower production costs. Although yield is always a concern, manufacture-time faults in image sensors pose a relatively minor challenge to the final application because imager chips can be readily calibrated before leaving the factory. Unlike digital devices, small numbers of defects can be tolerated in image sensors.

After calibration, only two solutions are available. The simplest solution is to discard sensors with excessive faulty pixels so that only high-quality imagers reach the product. Alternately, when the quantity and nature of defects is acceptable, the locations and types of defects can be stored in a memory, allowing the end application to deal with them intelligently. For example, manufacturers typically subject cameras to an optical calibration procedure in order identify faulty pixels before the product leaves the factory (see Section 3.1 for an example process) [32-37]. The location (i.e. pixel coordinates) of each fault is recorded in a fault-map, which is stored in the camera's firmware. During in-field operation, the DSC can therefore apply a correction algorithm to fix the resultant pixel value at each known-faulty pixel location in every image captured. Often, the correction is as simple as replacing the faulty pixel with a neighbour or an average of several neighbours.

Conversely, in-field defects, those created after the product leaves the factory, pose a further challenge: because in-field faulty pixels develop after factory calibration, the final application does not know about them and cannot correct for them.

A detailed review of individual in-field failure mechanisms in microelectronics is beyond the scope of this thesis(see [38-41] for more details). However, the cause of faults in imagers needs to be determined in detail. To this end, this section classifies defects into two categories, "material degradation" or "external stress," according to the source of the damage mechanism. Material degradation arises from intrinsic changes of the materials within the sensor array under ordinary operating conditions. External stresses are unexpected events that cause failures in good devices. The characteristics of these defect mechanisms, which are detailed in the section to follow, will be used in Chapter 3 to determine the underlying mechanism responsible for in-field faults in imagers. In both cases, a particular challenge of evaluating defect mechanisms in image sensors is their sensitivity to parametric faults that may otherwise be tolerated by digital systems.

2.5.3 Material Degradation

Material degradation refers to changes in electrical properties of any of the thin-film layers in a semiconductor device, including the silicon bulk, ion-implanted regions, gate dielectrics, gate materials, conductors, and inter-layer dielectric materials. Some common examples in standard CMOS circuits are gate oxide dielectric breakdown, ion migration, hot carrier degradation, and electromigration [38].

Briefly, gate oxide dielectric breakdown increases transistor gate currents, which contribute to signal leakage. Ion migration into gate oxides can lead to variations in transistor threshold voltages, affecting parameters like leakage drain currents, source-follower gains, or even APS reset levels. Hot carrier degradation occurs when electrons and holes, accelerated by large electric fields in MOSFETS, are injected into the gate oxide, where they can be trapped or cause defects.

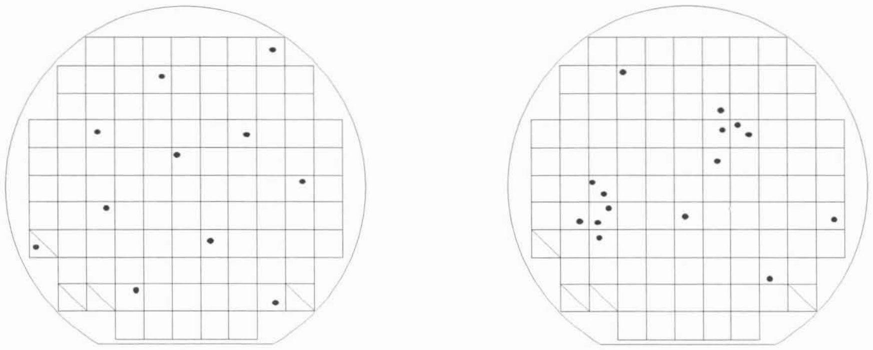
Consequently, the transistor experiences a threshold voltage shift with the associated effects on pixel characteristics. Electromigration is the movement of metal atoms in conductor lines, caused by the flow of electrons when large currents are applied. This displacement of metals can eventually cause open circuits (or short circuits in some instances) that would completely disable pixels[38].

While all of the above mechanisms are vastly different, they all affect the same family of materials in integrated circuits (ICs) and we can expect the resulting faults to share the following spatial and temporal traits:

- Multiple faults will form in clusters of closely-spaced failures localized to relatively small areas;
- Parametric faults will develop in a gradual and continuous process with the observed effects becoming progressively more pronounced over time;
- Faults will develop at an increasing rate, such that the fault density over time will grow faster than linearly.

The spatial distribution of faults can be directly related to the IC manufacturing processes because the initial quality of the resulting thin-film materials determines the susceptibility to failure of circuit elements and thus pixels. This is especially true for the very regular arrays found in imagers, where the local environment around each pixel is the same everywhere across the chip. Consequently, in-field material-related failures will follow the same statistics as manufacture-time defects that affect production yields. In particular, early yield models (incorrectly) assumed that defects develop independently across a wafer's surface, resulting in a uniform distribution as in Figure 2.17(a). To the contrary, fabrication engineers quickly found that defects tended to cluster into localized clumps, thereby affecting fewer die simultaneously, as illustrated in Figure 2.17(b)[41]. Thus, we can expect in-field material-related failures to form in similarly sized clusters. Given the large area sensors utilized in current high-end DSCs (i.e. about

2.5-times the area of the large modern digital microprocessors; see Chapter 3), these fault clusters would be readily localized to regions in a given sensor.



(a) Uniform defect distribution. *(b) Clustered defect distribution.*

Figure 2.17. Theoretical defect distributions across a single wafer. (a) Naïve uniform distribution, (b) observed distribution with clustering.

Some imager-related device failures can safely be treated as sudden events, as is typically done in the literature on general IC reliability. For example, electromigration may pass unnoticed until an open circuit develops and disconnects a pixel from the column bus. However, most of the physical degradation mechanisms are themselves continuous processes that gradually change device parameters over time. For instance, ions may accumulate over time in oxides, continually varying the threshold voltage and transconductance of the transistor. Similarly, injected hot carriers accumulate over time with similar effects. In digital devices, these mechanisms have little impact until sufficient damage accumulates to cause outright failure in a transistor (e.g. complete dielectric breakdown through a gate oxide), justifying the treatment of those failures as sudden events. However, imagers are sensitive analog circuits, and even small shifts in the parameters of the device in any of the imaging architectures could lead to noticeable changes in imaging characteristics. Thus we can expect that most faults, with the exception of a few catastrophic failures like the open circuit example above, would develop gradually over time. Parameters like sensitivity, offset, or dark current are all analog quantities that may be shifted by the accumulation of damage as the device materials degrade over time.

The failure rate of devices due to material-related catastrophic faults is well-known to be an increasing function of time (i.e. the number of failures per day increases with the age of the IC)[38]. This occurs because the continuous degradation of material properties makes an increasing quantity of devices susceptible to failure from ordinary applied stresses. However, the exact rate of increase will depend on the physical mechanism responsible for the observed failure. Similarly, the degradation rate of parametric faults will depend on the exact nature of the mechanism responsible such that the nature of the failure rate cannot be predicted ahead of time (note: in the case of parametric faults, we can define failure rate as the rate of change of parameters like sensitivity, etc). Many of degradation processes described in this section are positive feedback processes that will indeed lead to an increasing parametric failure rate, while many others may exhibit a nearly constant degradation rate. Nonetheless, few mechanisms slow down or reverse themselves in the conditions typical for image sensors [38].

2.5.4 External Stresses

External stresses typically refer to user or environment-induced damage, such as over-voltage power supplies or electrostatic discharge (ESD), which can rupture gate oxides and render sensors unusable. Similarly, packaging failures, like delamination of the die from the package frame, can also occur due to thermal stress or excessive moisture absorption[38]. However, in all such cases, the entire imager is disabled, making these mechanisms irrelevant to pixel-level faults.

Radiation is another external stress known to cause failures in all microelectronics in both terrestrial and space environments, although imagers are especially vulnerable because of their sensitive analog nature. Specifically, research on space-borne devices indicates that energetic particles like ions, protons, neutrons, photons, etc., create hard pixel-level failures via two dominant mechanisms: ionizing and displacement damage[42].

Displacement damage results when particles collide directly with atoms in the device, causing them to move. When this occurs in the silicon bulk, the regular structure of the silicon lattice is disrupted and additional energy states are created inside the bandgap[43]. As was described in Section 2.1, these trap states give rise to increased dark current when the defect is located in or near the photodetector depletion region. Laboratory experiments have demonstrated the creation of hot pixels as a result of proton and neutron irradiation in both CCD and APS arrays [6, 44-46]. Moreover, Chugg et al. confirmed that a single neutron interaction is sufficient to create a hot pixel [46], demonstrating that hot pixel effects are not limited to environments with a high particle flux. Accordingly, in low flux environments, radiation-induced displacement damage is likely to be confined to small areas (i.e. regions within a single pixel).

Both photogate and photodiode sensors perform photoelectric conversion and charge collection within the silicon bulk, making them both susceptible to dark current spikes induced by displacement damage. Consequently, all of the CIS and CCD architectures described in this thesis are sensitive to displacement damage-induced hot pixels. However, CCD sensors are potentially further affected because lattice defects in the transfer channels can cause reduced charge transfer efficiency (CTE).

Ionizing damage results when an energetic charged particle travels through dielectric layers (i.e. gate and isolation oxides) and deposits charge along its path. In many cases, the deposited charge is immobile and remains permanently in the oxide, causing threshold voltage shifts in MOS devices[47]. For example, accumulated positive charge in the CCD gate oxide will increase the depth of the depleted region and hence increase device sensitivity, as well as dark current. Moreover, charge accumulation in the gate oxides of transistors in an APS could lead to a shift in sensitivity if the source-follower or transfer-gate are hit. Ionizing damage also leads to an increase in trap states at the Si-SiO_2 interface, giving rise to increased dark currents in some photodetectors[48]. Similarly, electrically-neutral particles, like neutrons, can also deposit charge as a by-product of collisions with the device[49].

Note that several other, more destructive radiation-induced mechanism are also known to occur in space-borne microelectronics, including single-event latchup, single-event gate rupture, and single-event snap-back [50]. However, many of those effects are typically limited to dense digital circuits, and none of the detailed experimental radiation-effects studies have reported significant occurrences of such faults in CIS or APS arrays[12, 42, 44-46, 48, 51-53].

2.5.4.1 Cosmic Rays and Radiation Sources

Cosmic rays and radioactive emission from materials in the camera's environment are the primary sources of radiation for imagers. Cosmic rays are energetic particles originating from various sources in outer space. Primary cosmic rays in space are made up predominantly of protons and alpha-particles (i.e. Helium nuclei) with very high energies[54], which readily inflict damage on space-borne sensors.

Space-borne cosmic rays also impinge on the earth, interacting with the planet's magnetic field and atmosphere. As these energetic particles travel toward the surface, they collide with atmospheric particles, with each collision creating a dispersed shower of new, lower-energy particles. This cascade of showers produces ground-level cosmic rays, called Terrestrial Cosmic Rays (TCR), that are distributed relatively uniformly over large areas up to 100 km, although even small showers are much larger than a typical image sensor. Neutrons make up the most significant component of TCR, arriving at an attenuated but significant flux of approximately 10^3 neutrons/mm^2-year. Ziegler further estimates that this flux results in about 2.5 collisions per year between neutrons and active silicon in an average chip[54].

Note that both altitude and geographic location severely affect the flux and spectrum of terrestrial cosmic rays. Altitude effects arise because of the interactions between atmospheric and cosmic ray particles. Geographic location is a factor because the Earth's magnetic field, which shields the surface by deflecting particles, becomes progressively less effective near the planet's magnetic poles. For example, neutron flux at transcontinental airplane cruising altitudes (about 12 km) is approximately 100 times higher than at ground level and contains a slightly larger proportion of protons[54]. Similarly, mountainous regions or cities at high elevations, such as Denver, Colorado, experience higher cosmic ray densities than sea-level cities like Vancouver, BC. As an example, Table 2.3 shows the cosmic ray intensities from several locations relative to the author's location in Vancouver[55]. Note that the TCR intensity throughout this sampling of locations varies only by a ratio of 7.1 in the worst case.

Table 2.3. Relative cosmic ray intensities of some sample cities[55]

City	*Relative intensity*	*Elevation (m)*
Vancouver, BC	1.00	12
Calgary, AB	2.52	1048
Denver, CO	3.95	1610
Geneva, Switzerland	1.38	430
Hong Kong, China	0.56	33
Tokyo, Japan	0.63	4

Over time, TCR flux intensities at a particular location remain very stable. Variations caused by the changes in the solar cycle amount to than 20% worst-case difference over 40 years, with year-over-year variations limited to less than 5% [54].

Damaging radiation may also originate inside the camera system. The camera housing, lens glass, and even the cover glass used to shield the sensor have been suggested as sources of damaging radioactive particles. For example, trace amounts of radioactive thorium and uranium in the packaging materials of early dynamic memories contributed to significant and unexpected soft error rates in those devices[56]. However, recent experiments have concluded that cosmic ray neutrons are more likely to cause such damage[52].

2.5.4.2 External Stress Characteristics

With TCR established as the most prominent source of energetic particles that lead to radiation-induced external stress failures, we can predict the following spatial and temporal characteristics of such faults:

- Uniformly distributed faulty pixels across any given sensor;
- Faults limited to a individual pixels (i.e. no large clusters of faults);
- Sudden activation of faults arising after a successful particle interaction;
- Continuous and roughly constant rate of fault development.

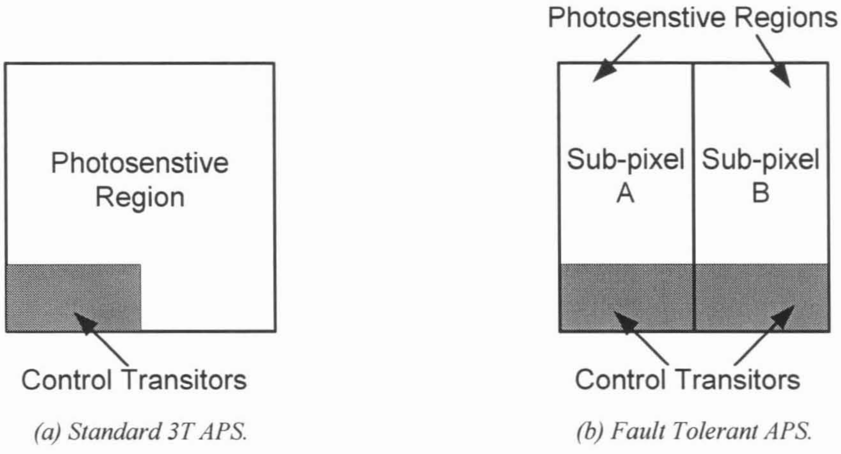
2.5.5 In-Field Fault Summary

This section has reviewed the basic expected behaviour of in-field faults in image sensors. The temporal and spatial characteristics determined for each class of defect mechanisms will be applied again in Chapter 3 to attribute a source to experimentally-located faults in commercial imagers.

2.6 Fault Tolerant Active Pixel Sensor

The Fault Tolerant Active Pixel Sensor (FTAPS) was proposed in 1999 by Chapman and Audet[15] to improve the manufacturing yield of image sensors. Originally intended to negate simple manufacture-time stuck-high and stuck-low faults, the design incorporates redundancy directly into the APS cell with minimum area overhead or control complexity.

The block diagrams of Figure 2.18 illustrate how this redundancy is achieved with minimal impact to sensing performance. A standard current-mode APS is split into two equal halves and the control transistors are replicated in each sub-pixel. The photosensor is also divided into equal components so that the overall photosensitive area remains the same. Because the active components occupy so little area in modern fabrication processes, the impact is a minor reduction in fill-factor compared to a standard pixel when the pixel dimensions are kept the same.



(a) Standard 3T APS. *(b) Fault Tolerant APS.*

Figure 2.18. Block diagram of circuit layout for (a) standard 3T APS and (b) FTAPS.

The schematic of the FTAPS in Figure 2.19 suggests how the two sub-pixels operate independently and in parallel. During integration, both sub-pixels collect and store light in their respective photodiodes. Once the common *row select* signal is activated, the current-mode outputs from both sub-pixels are mixed into a single common current-mode signal, I_{Out}. In this way, a fault in either sub-pixel is isolated to that side, while the other sub-pixel continues to produce a usable output. A detection and correction algorithm can then be used to extract the value measured by the working half. For example, to correct for a stuck-low sub-pixel fault, I_{Out} simply needs to be multiplied by 2.0 in the camera.

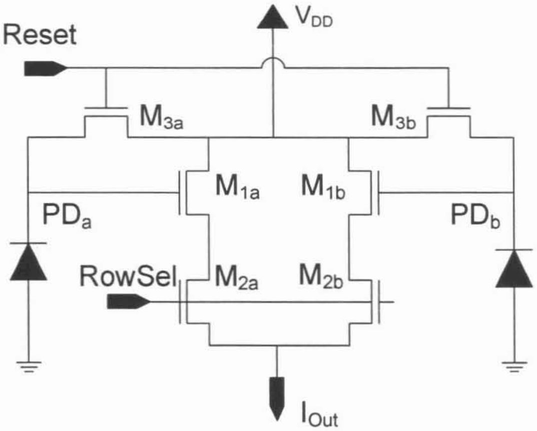


Figure 2.19. Schematic diagram of the FTAPS.

Because the control and output signals are shared between both halves, no complexity is added to the control and readout circuitry. Moreover, the shared nature of the pixels allows them to be placed closer together to sample the same incident light.

The FTAPS was previously implemented in 0.35 µm and 0.18 µm standard CMOS technologies and shown to exhibit sensitivity on par with standard 3T APS designs [16, 17, 20, 21]. Jung [20] and La Haye [21] also analyzed the noise performance of the FTAPS with and without defects and found the SNR output from the FTAPS is greater than or equal to the SNR from standard sensors without defects. When defective, the FTAPS SNR was at worst one half that of a working standard pixel but infinitely better than a faulty standard pixel.

These previous works focused on testing the FTAPS with simple stuck faults. Chapter 4 of this thesis extends that research by evaluating how the FTAPS behaves under the influence of more complex hot pixel faults, which Chapter 3 shows are the dominant fault type in commercial cameras

2.7 Summary

Both CCD and CIS imagers convert incident photons to charge via the photoelectric effect in the silicon bulk and then capture that charge in a reverse-biased photodiode or depleted photogate. Dark generation also occurs inside both photoelements and leads to an undesired dark signal that diminishes their continued light collection capacity. Active pixel sensors use a single-transistor source-follower inside the pixel to amplify signal from the photodiode for transmission to chip-level amplifiers and analog-to-digital converters. Charge-coupled device imagers shift charge to chip-level circuitry by controlling the surface potential of rows of abutting MOS capacitors. Both sensor types achieve a similar, linear illumination response, which is characterized by the sensitivity, offset, and dark current parameters.

Pixels that exhibit significant shifts in these parameters are called faulty. Fully stuck, abnormal sensitivity, and hot pixels are used to describe potential fault types. Defects are the physical points of damage that lead to faulty pixels. In-field faults occur when a defect develops after the camera leaves the factory. Physical mechanisms causing in-field failures can be categorized as material degradation or external stress, where the former is due to an intrinsic failure of the materials in an imager, while the latter refers to external event causing damage to otherwise good imagers. Material degradation mechanisms will be related to the fabrication process and will share its characteristics. Such faults will likely occur in large areas, forming clusters; they will develop gradually over time as the material fail; and their growth rate will increase over time. External stress-related failures will most likely be caused by radiation damage, whereby incident energetic particles collide with or pass through the semiconductor, damaging it. As such, these faults will likely occur at individual pixels, with fault locations distributed uniformly across as a sensor. In addition, radiation-related failures will develop suddenly and at a constant rate throughout the imager's lifetime. Chapter 3 will gather empirical evidence about the nature of in-field defects that really occur in common cameras to determine the prevalence of the defect types discussed so far, and determine which category of mechanisms best applies to common sensor arrays.

Furthermore, fault tolerant pixel designs have already been shown to handle manufacture-time fully stuck faults. Experiments in Chapter 4 will build on this previous research and show that the FTAPS can also combat the more complicated hot pixel fault type.

CHAPTER 3

IN-FIELD DEFECT CHARACTERIZATION

Camera users have long reported anecdotally that some pixels in their DSCs become faulty over time, even when operated in benign environments like a living room. However, such reports have usually been qualitative in nature, and details like how and when pixels fail, the number of failures, and the behaviour of failed pixels have not been available. There is also the reporting bias: only users experiencing failures report them while those without faults are silent. Although space-borne imagers and other research instrument cameras destined for harsh environments have been studied extensively, in-field pixel failures in terrestrial imagers have been seemingly ignored in the literature. One relevant study was conducted recently by Theuwissen [53], but his experiments were limited to a particular CCD sensor model and considered a simplified defect model by measuring only the magnitude of hot pixels.

This chapter describes experiments to determine the nature of in-field pixel failures in ordinary consumer digital cameras, with the aim of understanding the responsible defect mechanism and proposing a future solution. Laboratory calibration is performed on a set of digital still cameras to measure pixel response characteristics and determine how faulty pixels deviate from the norm. The quantity, location, and illumination response of pixel faults are then considered in order to quantify the following:

- Prevalence of in-field faults;
- Types of faults affecting real cameras;
- Rate at which faults develop;

- Long-term stability of pixel faults.

Finally, these measured parameters are subjected to statistical analysis to extract details about the spatial and temporal distribution of in-field faults in order to infer characteristics of the causing defect mechanism. Ultimately, the most probable mechanism is identified.

3.1 Experimental Method

Previous studies of aging in solid-state imagers specifically focused on hot-pixels, measuring and reporting only the distribution of dark currents in a given sensor[53], while ignoring other important details like spatial and temporal relationships between faults, as well as the illumination response of those faults. Furthermore, most aging experiments directly applied cumbersome laboratory tests to a particular sensor, limiting data to that particular implementation design and technology. The experiments described here build on that earlier work by collecting both spatial and temporal fault data while testing for all possible fault behaviour described in Section 3.1. Furthermore, the tests are designed to be performed on complete camera systems, allowing many different sensor designs to be considered in the same data set.

Darkfield (no illumination) and brightfield (uniform illumination) calibration of pixel responsivity form the core of the testing procedure. In addition, to verify that identified pixel faults are the result of in-field defects and not manufacture-time failures, photographs taken early in the lifetime of each camera were analyzed for the presence of faulty pixels. This processing step was a limiting factor in determining the criteria for labelling a pixel as faulty. Only those pixels that had a noticeable impact on a reasonable photograph (e.g. a 0.5 second exposure in the case of a hot pixel) were labelled as faulty.

A brief summary of the test procedure is as follows:

1. Capture darkfield images.
2. Capture brightfield images.
3. Analyze data and identify faults.

4. Compare fault map to early-lifetime images to isolate in-field defects.

These experiments were designed to simplify data collection from a large sample of commercially available DSCs. Because the focus is on in-field defects, only cameras that have been in the field for a significant amount of time can be considered. Therefore, to keep the cost and timeframe of this project reasonable, volunteer photographers were asked to calibrate their own cameras according to the specified procedure and submit the resulting data for analysis. As such, the procedures in the following sections were designed to balance accuracy with ease-of use because the typical camera user does not own complex optical apparatus.

3.1.1 Test Cameras

Testing was performed on high-end digital Single-Lens Reflex (SLR) cameras at various stages in their lifetimes. These cameras are professional or semi-professional grade, commercially available DSCs, selected because they allow manual control over most aspects of image acquisition so that almost any calibration process can be applied. Both CIS and CCD imagers from several technology generations were included. All tested sensors were relatively large area (about 3.5 cm^2) devices as is the state-of-the art in high quality cameras.

Further details about each camera are given in Table 3.1 Cameras are sorted according to sensor technology (CCD or CIS) and, when publicly available, the specific sensor topology is given in the third column. Age is the approximate time between the most recent testing of each camera and the time it was first used by the owner (which may differ from the manufacturing date). The shortest recognizable model name is listed for each camera.

Table 3.1. Details of tested digital still cameras.

			Number	Sensor	
			of pixels	dimensions	Age
Camera	Model	Sensor type	(millions)	(mm × mm)	(years)
A	Nikon D1X	Unnkown CCD	5.3	23.7 × 15.7	4.8
B	Nikon D50	PS-ITCCD	6.1	23.7 × 15.5	2.0
C	Nikon D70	PS-ITCCD	6.1	23.7 × 15.5	1.4
D	Nikon D80	PS-ITCCD	10.2	23.7 × 15.5	0.6
E	Nikon D80	PS-ITCCD	10.2	23.7 × 15.5	0.7
F	Nikon D200	PS-ITCCD	10.2	23.6 × 15.8	0.4
G	Canon EOS 300D	4T PPD APS	6.3	22.7 × 15.1	3.3
H	Canon EOS 10D	4T PPD APS	6.3	22.7 × 15.1	4.3
I	Canon EOS 350D	4T PPD APS	8.0	22.2 × 14.8	1.6
J	Nikon D2X	Unknown CIS	12.4	23.7 × 15.7	0.1
K	Nikon D2X	Unknown CIS	12.4	23.7 × 15.7	1.5

The primary benefit of using high-end DSLR cameras is the ability to read out raw image data. Separating the subtleties of pixel response from in-camera image processing is a particular challenge in characterizing commercial hardware, where only the pixel hardware characteristics are of interest, but the sensor functions as part of a larger system designed to create pleasing photographs. Consequently, pixel behaviour is distorted by processing that transforms raw image data into a recognizable picture.

For instance, some steps performed in the imaging pipeline of most high-end DSCs are linearization, (limited) dark signal removal, image scaling, demosaicing, exposure compensation, and compression. Linearization maps the non-ideal nonlinear pixel response (i.e. due to changes in pixel and floating diffusion capacitance) back to the ideal linear response. Basic dark signal removal eliminates the average background dark signal present in all pixels by estimating that offset from specially designated light-shielded pixels at the sensor periphery. Demosaicing is applied to images from all colour filter array cameras, where individual pixels are made sensitive to a small range of wavelengths (i.e. a pixel senses either red, green, or blue light). Demosaicing is the interpolation process that converts these individual colour samples into a complete red, green, and blue colour vector at each pixel site in the output image. Finally, compression processes, like the familiar JPEG method, are typically lossy, discarding some data from the image.

In short, all of these blocks in the imaging pipeline alter pixel values between the exposure and file-save steps, potentially enhancing or masking the impact of faults in an unknown way. Thus, it is absolutely necessary to use raw image data (prior to any processing) for characterizing faults. All measurements presented in the following section were obtained from linearized raw images prior to any geometric transformations or colour processing. Of course, each manufacturer may perform undocumented steps prior to saving raw data, which can be a hindrance to extracting meaningful measurements from captured data.

3.1.2 Darkfield Calibration

Darkfield calibration identifies hot pixels and any partially-stuck pixels with offset components. Hot pixels are identified by measuring the magnitude of dark current at each sensor site, while stuck pixels are found simply by locating pixels with exposure duration-independent offsets.

With the sensor kept in the dark, several images are captured at increasing exposure times. The dark current is then obtained by fitting a linear function to the results. Because of the temporal noise associated with dark current, curve fitting gives a more consistent result than the more commonly used method of sampling pixel values at long exposure durations. Figure 3.1 illustrates the results of a typical dark current fitting operation.

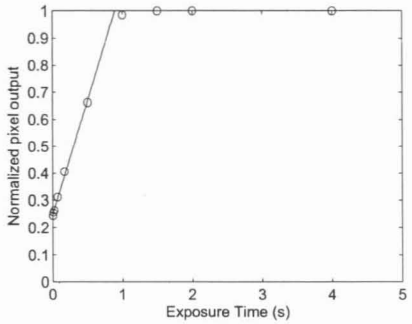


Figure 3.1. Measured pixel response from darkframe calibration with fitted linear curve.

During testing, 10-15 images were captured at exposure durations ranging from 1 ms to 6 s, and the camera gain was set to ISO 400 (see Section 2.4) to represent a typical photographing situation (photographers use this setting because output noise is typically negligible at ISO 400 and below on modern DSLR cameras). Keeping with the requirement that identified pixel faults should noticeably impact a photograph, pixels with dark current greater than 0.05/s (note the normalized units) were labelled as hot. Similarly, pixels with a dark offset greater than 0.10 were marked as faulty.

3.1.3 Brightfield Calibration

Stuck low and abnormal sensitivity faults require a more complex testing procedure because of their light-sensitive behaviour. The related llumination-dependant response parameters, such as pixel sensitivity and offset, are tested through brightfield calibration, in which the sensor is uniformly illuminated at several intensity levels (each one called a brightfield or flatfield) and the pixel output is recorded. Ideally, a linear function can then be fit to the data to determine the pixel's sensitivity and offset (terms m and Δ, respectively, in Eq.(2.11)). Pixels are deemed faulty when their sensitivity or offset deviate significantly from the rest of the population. To compensate for variations in spectral response, pixels of each colour are treated and compared independently.

However, generating a uniform light field and controlling its intensity is very challenging without specialized and expensive apparatus that are not commonly owned by even avid amateur photographers. Thus, a simplified test procedure was devised to enable any camera user to calibrate their camera with sufficient accuracy to identify most faults. Consequently, fault characteristics could be collected from a large sample, thereby increasing the statistical relevance of this study. The sample size is limited only by the willingness of users to contribute their time, enabling future extensions of this work to consider a broader range of cameras.

In this simplified method (details of which are given in Appendix A), users were asked to perform tests in an environment with relatively uniform ambient lighting, such as an office. With the lens removed to expose the sensor, a stack of diffusive and absorptive sheets (i.e. paper or similar material) were placed over the opening to create an even illumination across the sensor area (see Figure 3.2). The illumination intensity was controlled by varying the number of sheets in the diffuser stack. In this simple manner, about five uniform intensity levels could be generated without the need for any optical elements or the complexity of aligning a reflective diffuser.

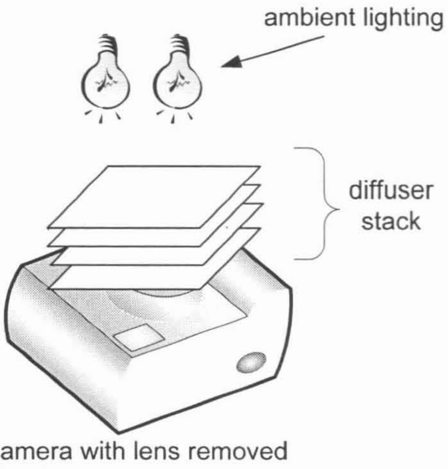


Figure 3.2. Experimental setup for simple in-field brightfield calibration.

Tests were performed with the camera lens removed to prevent lens defects and dust from influencing results. Very short exposure durations (10 ms or less) were used to avoid the accumulation of dark current. Camera gain was set to ISO 400 to simulate photographing conditions.

Figure 3.3 shows histograms of the pixel intensities in each colour channel of a single representative brightfield image captured using the stacked diffuser technique. The narrow peak in the distribution of all four histograms indicates that a reasonably uniform light field reached the sensor. Nonetheless, this method is clearly coarse, and fault detection was limited to pixels with large shifts in sensitivity (±20% or greater), stuck pixels, and pixels with large offsets.

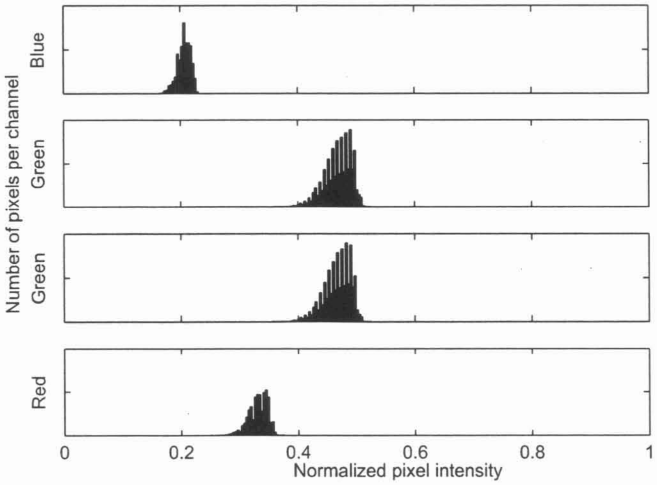


Figure 3.3. Sample histogram of brightfield values.

The coarse method was also augmented with a second technique whereby the output of each known faulty pixel (e.g. hot pixels) was compared to the mean of the outputs of the pixel's known-good neighbours of the same colour. Thus, fluctuations in the illumination could be smoothed by using good pixels as a reference, and the sensitivity of defects could be calibrated to a fine tolerance.

3.1.4 Temporal Data

Regular laboratory calibration is not feasible for the vast majority of cameras in users' hands. This led us to consider examining regular photographs captured throughout a camera's history. Because photographs are affected by faulty pixels in the same way as calibration frames, the presence of faults can be readily identified. Moreover, digital photographs store a wealth of information about the capture conditions, including the time and date the photograph was taken, the camera gain (ISO sensitivity), which lens was used, etc, which is all stored as metadata in most image files. Specifically, the Exchangeable image file format (Exif) defines how and where such information is stored in common image file formats like JPEG and TIFF, as well as proprietary raw file formats, which are often based on the TIFF standard. Thus, once faulty pixels are identified by calibration, their time of development can be determined by searching photographers' image collections for the earliest image that shows the influence of each fault.

The error in each development date is the period between the last known good image and the first image affected by the fault. Therefore, the accuracy of this search method is determined by the frequency of use and the habits of the photographer. For example, only long exposure images (greater than 1/30s) are useful for detecting many hot pixels, but photographers rarely shoot ordinary pictures at these speeds because of image blurring. Nonetheless, experience has shown the defect development date can often be determined with a margin of ±15 days or better.

3.2 Fault Characterization Results

Testing identified a multitude of in-field defects in most of the cameras tested. All faults were of the hot pixel type and some exhibited a partial offset as well (i.e. some were a combined hot-partially-stuck type). The light sensitivity of all hot pixels closely matched that of their good neighbours.

Despite their apparent prevalence in informal discussions of image sensor reliability, no fully-stuck (high or low) pixels were identified in any of the tested cameras. In addition, no pixels exhibited a significant shift in sensitivity. A detailed breakdown of the quantities and types of pixels found in each camera is given in Table 3.2.

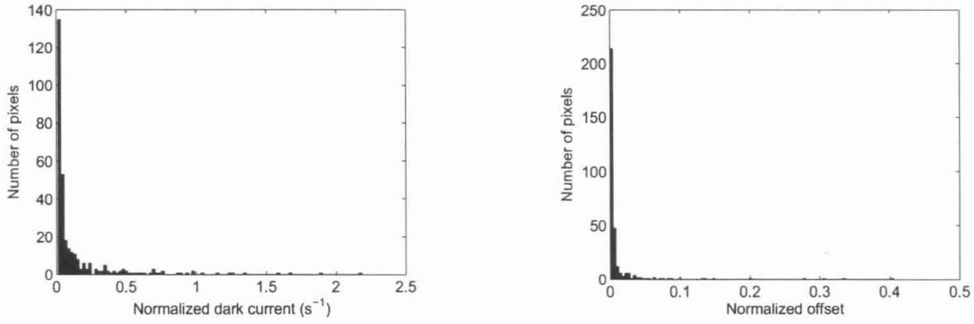
Table 3.2. Summary of faulty pixels identified in calibrated cameras.

Camera			*Quantity of identified faults*				
					Hot		
		Age		*Abnormal*	*Without*	*With*	
Name	*Sensor type*	*(years)*	*Stuck*	*sensitivity*	*offset*	*offset*	*Total*
A	Unknown CCD	4.8	0	0	26	0	26
B	PS-ITCCD	2.0	0	0	6	0	6
C	PS-ITCCD	1.4	0	0	17	0	17
D	PS-ITCCD	0.6	0	0	11	1	12
E	PS-ITCCD	0.7	0	0	11	0	11
F	PS-ITCCD	0.4	0	0	9	1	10
G	4T PPD APS	3.3	0	0	1	0	1
H	4T PPD APS	4.3	0	0	10	4	15
I	4T PPD APS	1.6	0	0	1	2	3
J	Unknown CIS	0.1	0	0	0	0	0
K	Unknown CIS	1.5	0	0	0	0	0

Note again that all of the identified faults were hot pixels. Moreover, the same types of defects were found in both CCD and CIS imagers and across all the technology variations of each type.

3.2.1 Fault Magnitudes

The dark current and dark offset magnitudes varied greatly from pixel to pixel in all of the sensors tested. Figure 3.4 shows histograms of both parameters, zoomed-in to highlight only the larger magnitudes. Both plots show that the vast majority of pixels exhibit very small dark current or offset while a small quantity show spikes in the parameters. The resulting distribution appears exponential, although no attempt has been made to quantify this. In short, the background dark current level remained low in all tested sensors, while a handful of faulty pixels showed extreme behaviour.



(a) Dark current.

(b) Dark offset.

Figure 3.4. Histogram tails of (a) dark current and (b) dark offset found in all hot pixels.

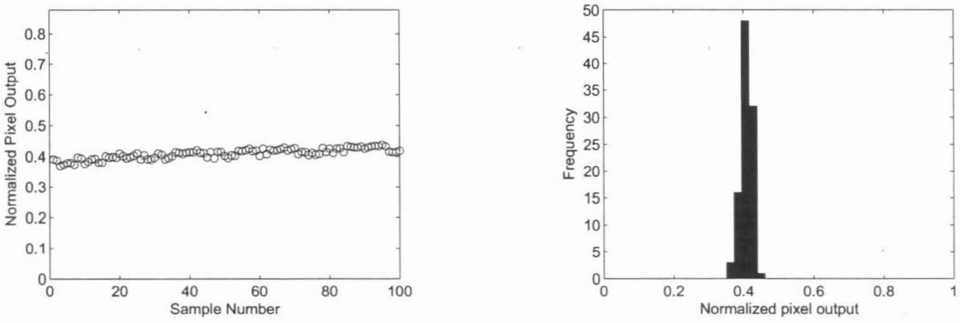
3.2.2 Random Telegraph Signal Faults

During testing, the magnitude of dark current appeared to fluctuate randomly in some hot pixels. To further investigate this phenomenon, 100 images were captured in rapid succession at the same exposure duration on Camera H. Images were taken at about 30s to 60s intervals to not excessively heat the sensor. The results (see Figure 3.6 on page 54) match the behaviour of *Random Telegraph Signal* (RTS) faults observed by other researchers during both proton [51, 57] and neutron[46] irradiation experiments.

For reference, Figure 3.5 shows the output values measured from a simple hot pixel, while Figure 3.6 and Figure 3.7 show the values measured from RTS hot pixels. Given that integration time is fixed, the offset due to dark current should remain constant and the recorded value should be the same in all frames. However, Figure 3.6 shows that the output appears to hop between two distinct values, suggesting that the dark current is randomly switching between two discrete levels. Similarly, Figure 3.7 shows that RTS defect hopping between four distinct states.

Histograms of the output values in all three figures show that the normal hot pixel output is confined to one narrow range, while the RTS faults alternate between two or four distinct ranges. The few values that fall between the two discrete levels can be explained by a dark current transition during the middle of an exposure.

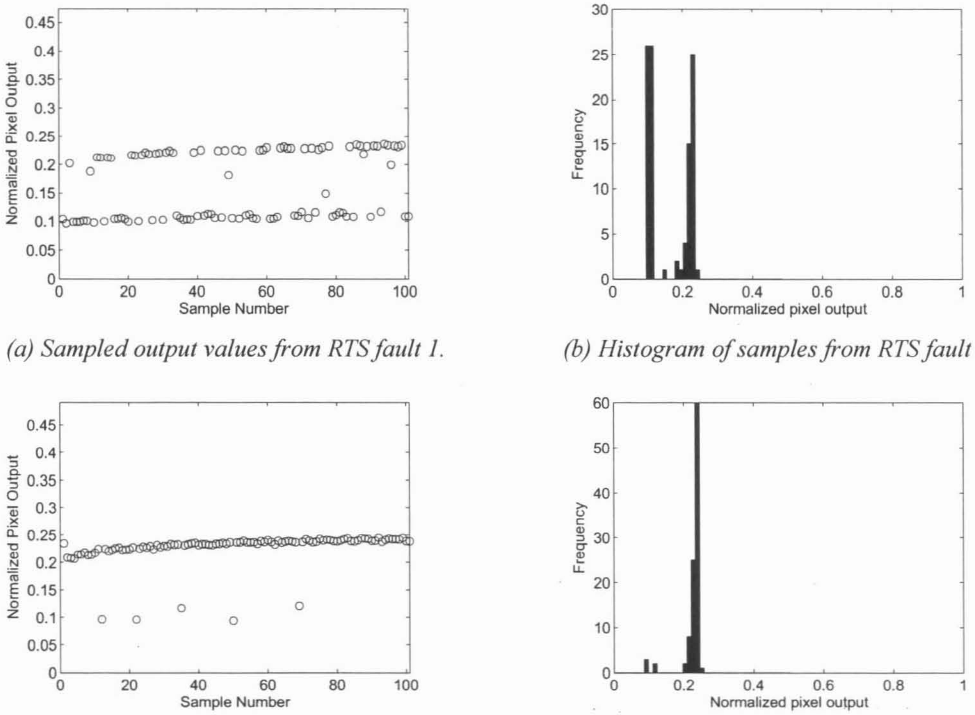
Of the 15 hot pixels in Camera H, 20% (i.e. 3 pixels) were identified as hopping hot pixels with a noticeable change in dark current. Because the transition between states appears to be random in all three pixels, the RTS fault can be described by three parameters: base dark current level, hopping amplitude, and average lifetime in the lowest state. Visual inspection of Figure 3.6 suggests that the lifetime can vary from under 30 seconds to several minutes. To obtain a more accurate lower bound on the lifetime, the above experiment was repeated several times for exposure durations from 0.25 s to 8 s (we cannot acquire frames at a faster rate from a standard DSLR camera). The lower bound on lifetime was taken to be the exposure duration at which a continuous range of output values were observed instead of two discrete values. In all cases but the small-amplitude state in the 4-state fault, the lifetime appears to be greater than 8s, while the minor state appears to have a lifetime on the order of 1s.



(a) Sampled values.

(b) Histogram of sample values.

Figure 3.5. Rapidly sampled output values and histograms of values from a simple hot pixel at a single exposure duration.



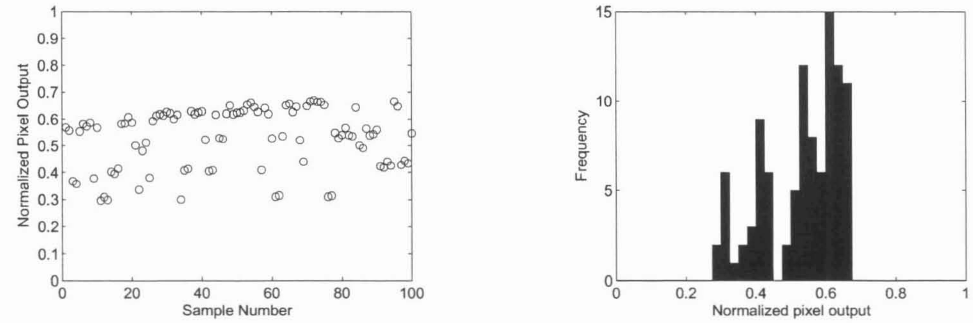
(a) Sampled output values from RTS fault 1.

(b) Histogram of samples from RTS fault 1.

(c) Sampled output values from RTS fault 2.

(d) Histogram of samples from RTS fault 2.

Figure 3.6. Rapidly sampled output values and histograms of values from 2-state RTS hot pixels at a single exposure duration.



(a) Sampled values from 4-state RTS fault.

(b) Histogram of samples from 4-state RTS fault.

Figure 3.7. Rapidly sampled output values and histograms of values from a 4-state RTS hot pixel at a single exposure duration.

The measured parameters for each RTS hot pixel are summarized in Table 3.3. The age of each fault (extracted in Section 3.3.3) is given in the final column to show that the RTS behaviour is not a transient anomaly in newly formed faults, which could stabilize over time.

Table 3.3. Summary of RTS hot pixel parameters.

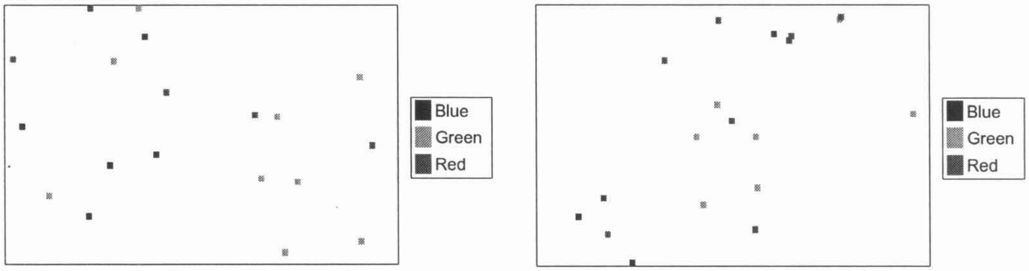
Pixel	*Base dark current (s^{-1})*	*Hopping amplitude (s^{-1})*	*Lifetime lower bound (s)*	*Number of levels*	*Defect age (days)*
1	0.12	0.20	> 8	2	1259
2	0.15	0.15	> 8	2	785
3	0.30	0.30	< 1	4	365

3.3 Fault Development Analysis

Faulty-pixel analysis has typically focused on the quantity of hot pixels and the distribution of their measured dark currents as seen in [53]. However, such an approach discards much of the information stored in each captured image, which could otherwise be used to gain further insight into the causing defect mechanism. In this section, traditional yield-analysis techniques that are usually applied to complete wafers are extended to in-field fault analysis in these large-area imagers. The coordinates of faulty pixels, which are readily found, are used to identify patterns in the spatial distribution of hot pixels and consequently the size of the underlying defect. Similarly, time stamps from captured photos are examined to determine temporal growth rates. All three characteristics are highly influenced by the nature of the degradation process and are examined in further detail in Section 3.4.

3.3.1 Fault Spatial Distribution

Detailed maps showing the positions of in-field faults across the sensor were created for all tested cameras. Figure 3.8 shows two such plots from representative cameras. On visual inspection of Figure 3.8, the faults appear to be uniformly distributed without any obvious clustering in localized areas.



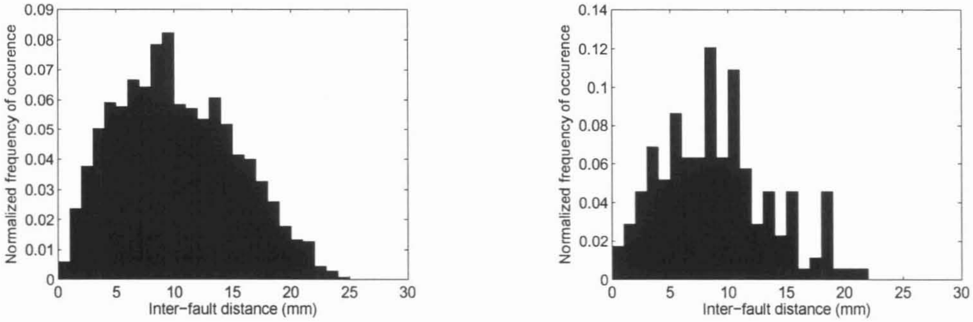
(a) Camera C.

(b) Camera H.

Figure 3.8. Defect maps showing the position of in-field hot pixels in two cameras.

Most significantly, no two faults developed adjacent to one another. That is, at each identified hot pixel, all eight neighbours were carefully characterized, and all of the neighbours were found to be good pixels. Thus, only a single pixel appears to be affected by each individual defect that causes hot pixels.

However, due to the small number of hot pixels observed in any given sensor, traditional statistical methods like the Chi-squared test cannot be applied directly to the defect locations to verify the uniformity of their spatial distribution. Instead, the distribution of inter-defect distances is examined. The distances between every pair of defects within each sensor are computed and are then compiled into a single distribution for all the cameras of each imaging technology. Histograms of the resultant distributions, shown in Figure 3.9, appear to be roughly uniform and exhibit no peaking toward short inter-defect distances.



(a) CCD Cameras.

(b) CIS Cameras.

Figure 3.9. Distributions of inter-fault distances for tested (a) CCD and (b) CIS cameras.

In fact, descriptive statistics for the distribution (see Table 3.4) indicate that mean CCD-imager inter-fault distance is about 10 mm or one third of the distance across a sensor chip (33% of the sensor diagonal dimension). Moreover, if defect clustering were prevalent, defects would be more likely to develop closer to one another and we would expect to observe multiple distinct peaks in the histograms. Peaks at small separations would arise due to narrow fault spacing within a cluster, while other peaks at larger separations would likely arise due to the spacing between distinct clusters.

Also, note the similarity between distributions from CCD and CIS imagers, despite the difference in available data.

Table 3.4. Summary of inter-fault distance statistics.

	Inter-fault distance statistic	
Technology	*Mean (mm)*	*Standard deviation (mm)*
CCD	8.94	4.66
CIS	10.20	5.08

For comparison, a simple Monte Carlo simulation was performed to estimate the expected inter-defect distribution for cases without clustering (clustering simulation models typically do not reflect physical processes and depend on additional parameters so they were not considered here). Defects were randomly placed in a simulated sensor according to a uniform distribution and the inter-defect distances measured. An average defect density of 16 defects per sensor was used (to match the average case for CCD imagers) and the experiment was repeated for 200 sensors. A histogram of the cumulative inter-defect distributions is given in Figure 3.10. As expected, the uniform random model generated an even distribution of defects as was observed in the real sensors. A Pearson-χ^2 comparison of the observed and simulated distributions for CCD sensors gives an excellent χ^2 value of 5.05, compared to a critical value of 27.6 (for 0.05 significance level).

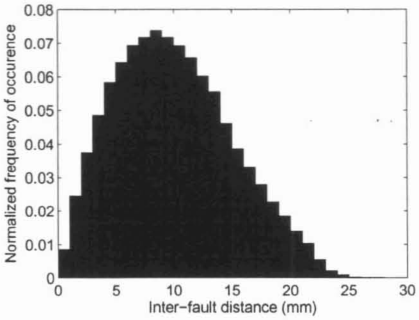


Figure 3.10. Simulated distributions of inter-defect distances with uniform random model.

In summary, the above analysis strongly indicates that hot defects can be treated as uniformly distributed across large area image sensors. With reference to the two classes of potential defect mechanisms, material degradation and external stress, this evidence conflicts with the expected behaviour of material degradation-related defects while supporting the possibility of radiation-induced damage. In particular, interaction between individual TCR and sensor particles can be treated as independent events, which gives rise to a uniform defect distribution across any given sensor.

3.3.2 Defect Size Estimate

The absence of multiple-pixel faults (e.g. only a single pixel is observed as hot while all neighbours function properly) strongly indicates that the size of the defects leading to hot pixels is much smaller than the dimensions of a single pixel. While the exact size of the defect cannot be measured directly, a straightforward statistical analysis of the existing data shows that hot pixel defects are indeed very small and almost point-like.

We begin by modelling the imager as an array of square pixels with dimensions $w \times w$ µm, as shown in Figure 3.11. Hot pixel defects are treated as circular spot defects with some random radius, r. In this first order approximation, the entire pixel area is assumed sensitive to defects such that any overlap between the defect area and a pixel will lead to a hot pixel fault. Because of the similarities between CCD and CIS imagers in developing hot pixels despite the significant difference in circuit layout, this first assumption seems well justified.

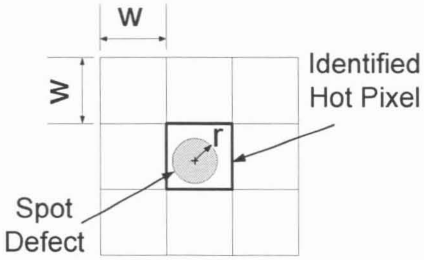


Figure 3.11. Sensor and defect model.

A hot pixel without faults in adjacent pixels is called an *isolated fault*. When a defect lands in the pixel area, it causes an isolated fault if the entire defect area remains within the $w \times w$ boundaries of the original defective pixel. Conversely, a cluster of faults is created if the defect area extends into any neighbouring pixels. Therefore, an isolated fault develops if the centre of a defect with radius r is within the hatched *Safe Area* shown in Figure 3.12.

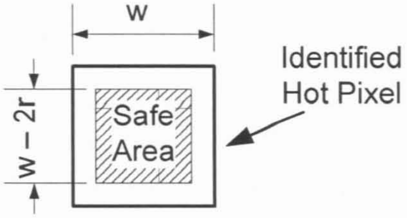


Figure 3.12. Safe region of the pixel area that leads to isolated faults.

Assuming that defects are uniformly likely to develop anywhere in the pixel, the probability of a defect with random radius, R=r, causing an isolated fault is thus given by

$$\Pr(Isolated\_Fault \mid R = r) = \begin{cases} \left(1 - \frac{2r}{w}\right)^2 & r < \frac{w}{2} \\ 0 & r \geq \frac{w}{2} \end{cases}. \tag{3.1}$$

We now consider all N identified hot pixels and make the simplifying assumption that all of these faults have been caused by defects with the same radius, R=r, which is justified by the final result that the defect size is indeed very small. Therefore, the probability of observing only isolated faults at all N hot pixels is given by

$$\Pr(N\_Isolated\_Faults \mid R = r) = \begin{cases} \left(1 - \frac{2r}{w}\right)^{2N} & r < \frac{w}{2} \\ 0 & r \geq \frac{w}{2} \end{cases}. \tag{3.2}$$

Finally, Bayes' theorem can be applied to obtain on upper bound on the hot defect radius, R_{max}, according to the following expression:

$$\Pr(R \le r_{\max} \mid N\_Isolated\_Faults) = \frac{\int_{0}^{r_{\max}} \left(1-\frac{2r}{w}\right)^{2N} f(r)dr}{\int_{0}^{w/2} \left(1-\frac{2r}{w}\right)^{2N} f(r)dr}, \tag{3.3}$$

where *f(r)* is some assumed probability distribution function for the defect radius. Without further knowledge about the exact nature of the defects and their sizes, several simple but common distributions can be tried. In the simplest case of the uniform distribution, all defect radii on the range [0, $^{w}\!/_{2}$] are assumed equiprobable, giving the distribution in Eq. (3.4).

$$f_{Uniform}(r) = \begin{cases} \frac{1}{^{w}\!/_{2}} & 0 \le r \le {^{w}\!/_{2}} \\ 0 & otherwise \end{cases}. \tag{3.4}$$

Although the implied lower bound of zero in Eq. (3.4) appears to unreasonably suggest the existence of zero-radius defects, this simplification allows us to consider infinitesimally small point-defects. Conversely, the fixed upper bound was chosen because the values greater than *0.5w* will not influence the results due to the limits of Eq. (3.1).

To better align with more realistic defect models used in manufacturing yield analysis [58], an exponential distribution of defect radii is also considered in the following expression:

$$f_{Exponential}(r) = \begin{cases} k \cdot e^{-k \cdot r} & r \ge 0 \\ 0 & otherwise \end{cases}, \tag{3.5}$$

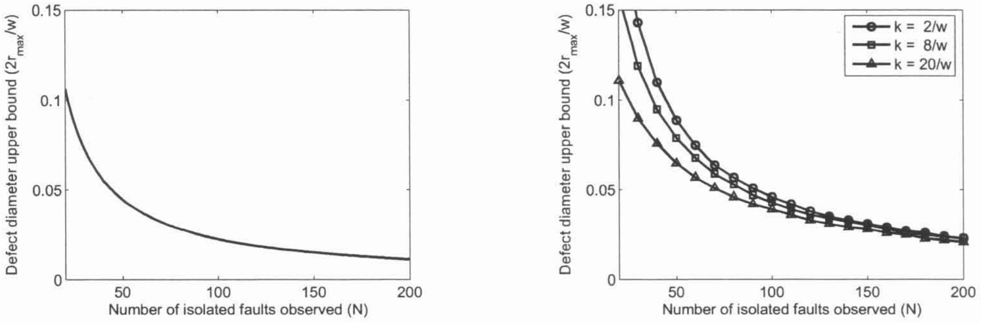
where, the parameter *k* is allowed to vary according to the desired spread in radii.

To establish an upper bound on defect diameter with some confidence level, *CL*, we set $Pr(R \le r_{max} \mid N\_ISOLATED\_FAULTS) = CL$ in Eq. (3.3), substitute the desired *f(r)*, and solve for the upper bound, r_{max}. A conservative approach is taken in the following calculations by setting

CL to 99%. The defect size upper bound is calculated as a relative value, $2r_{max}/w$, called the maximum normalized defect diameter, which allow the results to be applied to sensors with any pixel size.

The plot in Figure 3.13(a) shows the resulting $2r_{max}/w$ for increasing values of N when a uniform prior defect size distribution is considered. When the number of isolated hot pixels is large (i.e. greater than 100), we can be 99% certain that the diameter of all hot defects is less than 2.3% of the pixel width.

Similar results are obtained when considering a more realistic exponential prior distribution of defect sizes. Figure 3.13(b) plots the defect size upper bound for several values of the parameter k and increasing quantities of observed isolated faults. Small values of k (i.e. $k=2/w$) describe a nearly uniform distribution, while larger values of k (i.e. $k=20/w$) describe a highly skewed distribution where small defects are common and large defects are highly improbable. Once a large number of isolated hot pixels have been observed, results using all three exponential distributions converge. At 100 observed hot pixel faults, we can be 99% certain that all of the observed isolated faults were created by defects whose diameter is less than 4.5% of the pixel width.



(a) Uniform. *(b) Exponential.*

Figure 3.13. Upper bound on relative defect diameter assuming (a) uniform and (b) exponential distributions of possible defect sizes.

The fact that all of the tried distributions result in very small values of r_{max} strongly indicates that hot defects are point-like and supports the assumptions made above. Given that testing has identified 92 in-field hot pixels, all of which are isolated faults, $2r_{max}$ is conservatively 4.2% of the pixel width, and each defect occupies less than 0.15% of the pixel area. The tested sensors had an average pixel size of 6.9 μm × 6.9 μm, therefore the hot defect radius upper bound is less than 0.15 μm, and the defect area occupies less than 0.07 μm^2.

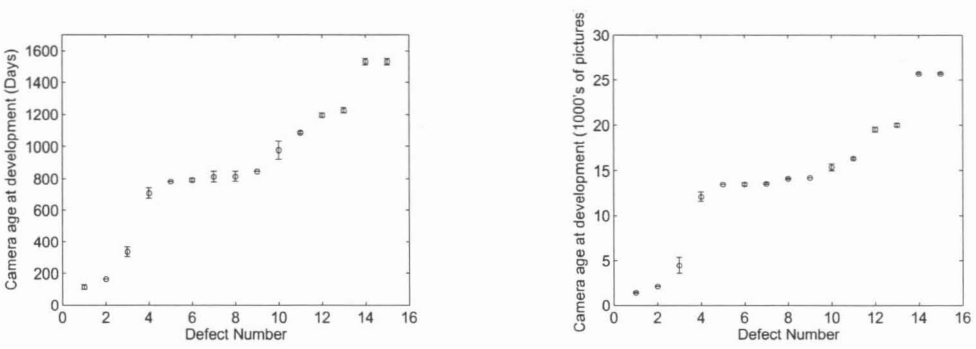
Once again, this evidence favours radiation-induced damage as the root cause of hot pixel faults. The single-event nature of cosmic-ray impacts and their limit area of effect aligns well with the observed point-like nature of hot pixels. On the contrary, material degradation would be expected to affect several pixels in any neighbouring region.

3.3.3 Temporal Characteristics

A key result of repeated darkframe analysis and defect history tracing through regular photos is that hot pixels appear static over time. Faulty pixels appear to develop suddenly, rather than gradually "turning on" over an extended period, and they do not heal over the durations tested in these experiments.

Camera H was the only camera with a sufficiently detailed photographic history available to trace the growth of all hot pixels over its lifetime, and this representative camera offers an excellent prototype for developing and testing some defect analysis techniques. Therefore, despite the limited statistical significance of the data, the following section demonstrates the effectiveness of the defect history analysis in viewing defect growth from a novel perspective.

The date when each defect developed in Camera H is plotted in Figure 3.14(a) along with error bars showing the accuracy of each measurement. The number of pictures captured by Camera H at the time when each defect developed is plotted in Figure 3.14(b). Large errors in the development date reflect either the availability of appropriate images (i.e. long exposure photos) or lulls in photographic activity by the user.

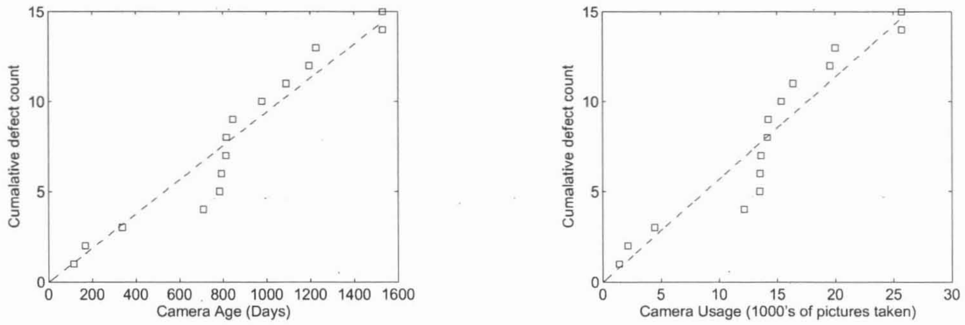


(a) Camera lifetime.

(b) Camera usage.

Figure 3.14. Development date of hot pixels in Camera H as a function of (a) Camera lifetime and (b) camera usage.

The above results are plotted again in Figure 3.15 to more clearly show a constant defect growth rate throughout the lifetime of the camera. Although the total quantity of defects is relatively low, these figures suggest the growth could continue indefinitely until image quality is significantly impacted. The data also shows a reasonable fit to a linear regression as indicated by the dashed lines in Figure 3.15. Defect growth rate is about 3.2 defects per year, or 0.53 defects per 1000 pictures.



(a) Camera lifetime.

(b) Camera usage.

Figure 3.15. Development rate of hot pixels in Camera H as a function of (a) Camera lifetime and (b) camera usage.

The roughly constant defect growth rate suggests that defect growth can be modeled by a Poisson process in which defects develop independently of one another at unrelated intervals. A simple Pearson χ^2 can quantify how well this model fits the observed data. The expected inter-defect intervals are modeled by an exponential distribution with parameter $\lambda = 1/102\ days$ and collected into three ranges with roughly equal expected frequency of occurrence. Table 3.5 shows the expected and observed number of inter-defect intervals falling into each range. The resulting χ^2 statistic,1.2, is much less than the critical value of 6.0 (for a 0.05 significance level), giving quantitative support to the hypothesis that inter-arrival times follow an exponential distribution and that defects develop independently.

Table 3.5. Distribution of defect arrival times for χ2 test.

Interval between defects (days)	0 – 41.4	44.4 – 112	> 112
Expected frequency	*5*	*5*	*5*
Measured frequency	*6*	*4*	*5*

While the above data set is admittedly limited, it serves several very useful purposes. First, the data clearly illustrates that defect growth is continuous. Second, preliminary analysis suggests that defects develop independently. Finally, the above investigation demonstrates that image history analysis can provide accurate information about the development rate of in-field defects from a viewpoint that has not been considered elsewhere.

Furthermore, these temporal characteristics contribute further evidence toward discerning the responsible mechanism. The "sudden turn-on" nature of hot pixel faults is consistent with the single-event nature of TCR interactions but completely at odds with the gradual degradation expected from material-related failures. The nearly constant fault development rate (with statistical variations associated with the small sample of faults) also supports TCR as the causing mechanism because of the steady flux intensities observed over long time periods. Variations in hot pixel development rates can be directly related to exact geographic location of a particular camera, which affects the incident TCR flux, as well as the availability of local shielding that may further attenuate incident particles.

3.4 Interpretation

In summary, hot pixels were the dominant fault type and no abnormal sensitivity or fully stuck pixels were found. Combining the results of the above analysis, radiation-induced external stress damage is the most likely cause of these faults, which presented the following characteristics:

- Uniform spatial distribution across sensors;
- Small and point-like with impact limited to a single pixel;
- Instantaneous activation from good to faulty;
- No change in fault characteristics following activation;
- Constant growth rate over time.

These attributes are all consistent with terrestrial cosmic ray (TCR) radiation damage but are opposed to the expected behaviour of material degradation, leaving the former as the most probable cause.

Moreover, faults exhibited the same characteristics in both CCD and CIS imagers and across several technology generations in both classes, indicating a common defect mechanism is responsible. In addition to the list above, all hot pixels shared the same basic illumination response in which sensitivity remained unaffected, and no other fault types beyond hot pixels were observed. Therefore, we can reasonably assume hot pixels in both CIS and CCD sensors are caused by the same mechanism or a family of closely-related mechanisms, which makes material degradation unlikely because the differing technologies and architectures rely on slightly different materials and processing techniques. Thus, an external stress, like radiation damage, is a more likely cause.

We can extend this reasoning to further isolate the responsible defects. First, as noted above, the similarity in fault behaviours indicates a common defect is responsible in all of the tested imaging technologies. Because only single-pixel faults were observed instead of column-wide or row-wide faults, the defect must occur inside each individual pixel. Moreover, given the wide variations in pixel architectures, the damage is most likely isolated to the photosensing element, which is similar in all designs. In particular, silicon displacement damage is the most likely mechanism because it can drastically increase dark current without affecting other imaging parameters, like sensitivity.

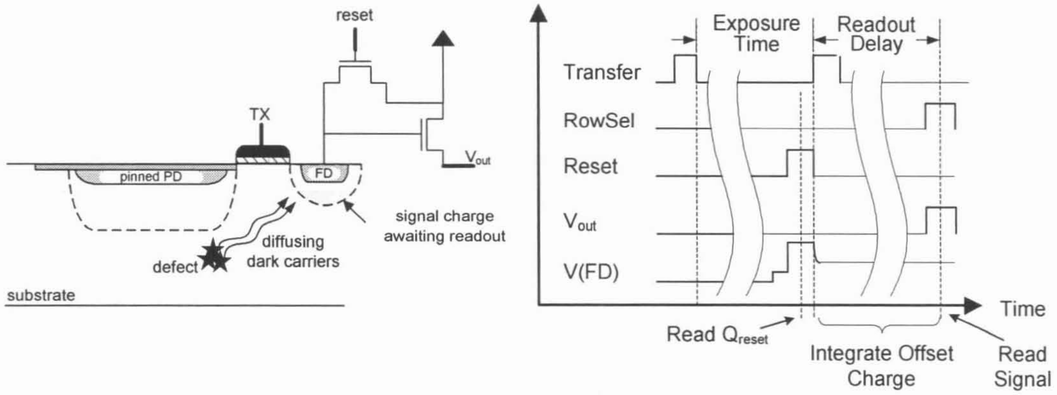
Terrestrial cosmic rays remain the most likely source of this radiation because of the apparently random and independent arrival of defects at a low rate. In fact, if the neutron collision rate predicted in Section 2.5.4 is repeated for image sensors with an active area of 350 mm^2 and an active depth of 10 μm, a defect development date of 5.5 defects/year would be expected. This rate is well in the range observed in the cameras tested in this experiment. Variations in defect growth rate between cameras and users can be readily explained by variations in the sensitivity of different designs to defects and variations in cosmic ray flux around the globe. For example, the depth of the photosensing element's depletion region and the relative size of the photosensing element will strongly influence how much dark current is collected, but these parameters are not made public by camera manufacturers.

Furthermore, the cameras with the fastest defect growth rates are known to have been taken on numerous high-altitude air flights, and have thus been exposed to larger amounts of radiation. For example, as a first order approximation, a return flight from Vancouver to Hong Kong would expose a sensor to the equivalent of 108 Vancouver-days of cosmic radiation.

3.4.1 Partially-Stuck Faults

While the bulk displacement damage clearly explains the creation of hot pixels, the source of partially-stuck sites is less obvious. Nonetheless, the described mechanism is consistent with that offsets observed at these faults, although the precise method varies for each of the three major imaging technologies considered. In all cases, an exposure duration-independent offset can only develop during a fixed-time interval of the imager operation.

The case of CMOS image sensors is straightforward because the 4T CMOS APS with pinned photodiode is the dominant pixel architecture[59, 60]. After the exposure duration in such cameras, the collected charge in all photodiodes is simultaneously transferred to their respective storage nodes (i.e. the floating diffusion – see Section 2.2.2) to await readout. During this waiting period, called the *readout delay*, dark current generated outside the photodiode can also diffuse to the floating diffusion to be captured there, as illustrated in Figure 3.16(a). The readout delay, which is illustrated in Figure 3.16(b), can be significant for pixels at the end of the read-out queue, leading to relatively long integration times and thus large offsets. For example, consider that most DSLR cameras cannot capture more than 10 frames per second, suggesting that dark current can be accumulated for up to 100 ms. Because the readout delay for a given pixel is independent of exposure duration, partially-stuck faults develop with fixed offsets.



(a) Physical mechanism. *(b) Timing diagrams.*

Figure 3.16. (a) Physical mechanism and (b) readout timing leading to partially-stuck faults in 4T CMOS APS imagers.

In addition, many sensors employ anti-blooming circuitry that is disabled during readout to prevent alteration of the signal charge[31], which would increase amount of collected offset charge relative to the exposure-dependant dark current. The exact offset magnitude will depend on the dark current magnitude, defect location within the pixel (i.e. dark current generated nearer to the FD is more readily collected), and the pixel's position in the readout sequence.

Determining the source of partially-stuck offsets in CCD imagers is complicated by the presence of two competing architectures (ITCCD and FFCCD) and the reluctance of manufacturers to reveal the particular design used. The principle of ITCCD offset is very similar to that of the 4T APS described above: after light integration, signal charge is transferred to a storage node, the VCCD, where it waits to be transferred to the readout node. However, in Progressive Scan ITCCDs (PS-ITCCD), signal packets do not dwell for a long time in any single VCCD well because they are shifted immediately after the HCCD is emptied, making it unlikely that any individual signal packet could be significantly affected. Conversely, in the more common Interlaced Scan ITCCDs (IS-ITCCD), all of the pixels are still exposed simultaneously but the array is read out one half-frame at a time[31]. Thus, the readout delay for every other pixel is at least half of the complete frame readout time, which increases the likelihood of accumulating a fixed offset in the same manner as the 4T APS.

In Full Frame CCD-equipped cameras, all of which must be equipped with a mechanical shutter, the fixed offset can potentially accumulate prior to exposure. When a frame capture is initiated, immediately before the shutter is opened, the CCDs are rapidly shifted out to empty any dark or residual charge. The finite duration of this rapid reset process and the delay before the slow mechanical shutter opens can combine to create a sufficient interval in which a fixed quantity of dark current can be accumulated in each CCD. Again, the exact magnitude of the offset will depend on the dark current magnitude and the pixel's position in the readout sequence. Note that a similar scenario is plausible in PS-ITCCD sensors.

3.5 Summary

Several high-end digital still cameras were put through calibration procedures by end-users to identify in-field faults. Cameras at varying ages with both CCD and CIS sensors were included. Darkfield calibration was used to find hot pixels and stuck high pixels, and a simplified brightfield calibration test was used to locate pixel faults with large sensitivity shifts. Almost all cameras were found to have developed significant quantities of hot pixels, some of which showed an offset in the illumination response, but no other faults were located.

Analysis of the distances between faults shows that hot pixels developed at uniformly random locations across all sensors. Furthermore, the absence of multi-pixel fault clusters suggests that hot defects can be treated as spot defects with less than 0.15 μm diameter. A detailed analysis of photographs from one representative camera shows that hot pixels develop continuously and suddenly throughout the lifetime of the camera with a rate of about 3.5 faults / year in this case. Together, these characteristics indicate that hot pixels in commercial cameras are caused by an external stress, particular displacement damage from terrestrial cosmic rays, rather than material degradation or user-induced damage.

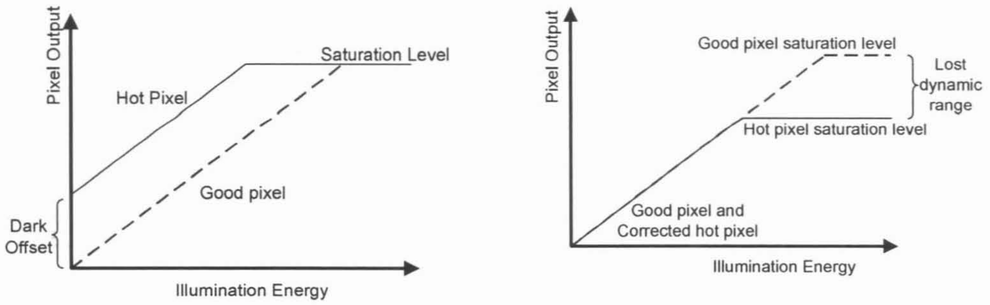
Because complete shielding is likely ineffective against the cosmic ray neutrons causing this damage, the following chapter describes how the fault tolerant APS architecture can be used to mitigate the effect of hot pixels in the field without sacrificing dynamic range and image quality.

CHAPTER 4

FAULT TOLERANT ACTIVE PIXEL SENSOR

The previous section clearly demonstrated that hot pixel faults are widely prevalent in many cameras and their quantities only increase with age. While a handful of faulty pixels in a two-year-old photo-camera may not be a cause for concern, 30-100 hot pixels are likely to impede the functioning of a five-year-old machine vision system. Moreover, image processing steps like demosaicing are known to enhance and spread the influence of hot faults over many pixels, further degrading image quality. Thus, a methodology is required to mitigate the impact of hot pixel faults in the long run.

Conventionally, darkframe subtraction is employed to "erase" hot pixels from pictures: following each image, a darkframe image with the same exposure duration is captured to estimate the dark signal at each pixel, and this value is subtracted from the original photo to remove the bright spots due to hot pixels. Although this simple method appears effective, it does not restore the lost dynamic range of the sensor because the accumulated dark current drives hot pixels into early saturation at lower than expected light levels, preventing them from measuring light accurately. For example, consider how a hot pixel's illumination response curves, shown in Figure 4.1, are truncated by darkframe subtraction, which thereby reduces the dynamic range and thus usability of sensors once they are damaged by hot defects. Simple darkframe subtraction ignores this effect and discards this region of dynamic range.



(a) Initial illumination response curves.

(b) Curves after darkframe subtraction.

Figure 4.1. Illumination response characteristics of hot pixels (a) before and (b) after darkframe subtraction.

Given the shortcomings of existing software-only darkframe subtraction methods, the aim of the work presented in this chapter is to develop a hardware method to alleviate the impact of hot pixel damage without sacrificing dynamic range. Experimental evidence from the previous chapter strongly indicates that bulk displacement damage to the photodiode elements is the cause of hot pixels. However, if cosmic ray impacts are indeed the source of these defects, such damage would be extremely difficult to prevent because of the penetrating nature of the impinging particles. Moreover, even if the above conclusions are incorrect and other damage mechanisms are responsible, modifying the pixels' materials properties may eliminate hot defects but would incur very expensive modifications to the manufacturing process.

Previously, a fault tolerance method, based on the Fault Tolerant Active Pixel Sensor (FTAPS) architecture (see Section 2.5.5) was created to reduce the impact of regular defects. In this thesis, we examine the FTAPS as a low-cost technique to mitigate the impact of hot pixels. Similar to how simple stuck faults are handled, the FTAPS can quarantine hot defects to one portion of the pixel while the remaining circuitry operates as intended. Faults are then identified by a simple darkframe test and a software algorithm corrects their output values, eliminating the offset due to the hot portion. The primary cost of this fault tolerance is a small reduction in fill factor and the necessity of contending with the non-linear response of the FTAPS, which operates in current mode.

A more detailed description of the hot fault tolerant pixel is presented next, after which the recovery algorithm is fully described. The effectiveness of this tolerance method is then investigated experimentally by emulating hot pixel behaviour in fabricated FTAPS arrays. First, an optical method is used to induce additional dark current in one sub-pixel, simulating deeply located displacement damage. In a second experiment, specially designed test pixels generate dark current that would be induced by damage located nearer to the surface. In both cases, the accuracy of recovered data is used as the metric for the viability of this scheme.

4.1 Theory of Operation

For convenience, a schematic diagram of the FTAPS is repeated in Figure 4.2. Recall that the device mitigates defects by quarantining their effects to a single sub-pixel. In the case of hot pixels, the two photodiodes independently collect all charge generated in their respective depletion regions, including both photocurrent and dark current. Therefore, the excessive dark current induced by the hot defect will deplete only one half of the complete pixel while the other half continues to operate unimpeded.

In short, the combined output of a partially defective FTAPS is always the sum of the good signal from the working half and some error value, such that an accurate image can always be recovered. Most importantly, the correction method is useful even when some sub-pixels have saturated from excessive dark signal. The complete dynamic range of the original pixel is preserved and sensitivity is only reduced by a factor of 2 in the worst case

Now consider the impact of a hot pixel defects on an FTAPS pixel. Given the miniscule estimated size of the hot pixel defect, as shown in Chapter 3, defects will be confined to one of the two halves of the FTAPS, making the above operating mode the most common. However, should multiple defects affect a single pixel, the FTAPS behaves simply behaves like a hot standard pixel, and performance is not lost over the standard pixel architecture.

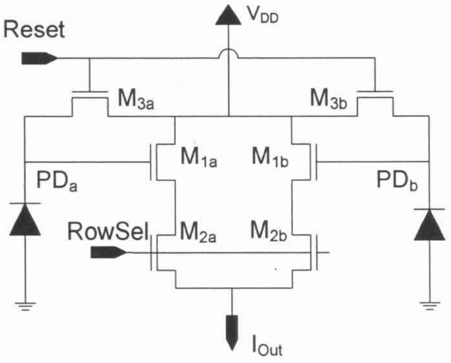


Figure 4.2. Schematic diagram of the FTAPS.

Now consider the behaviour of an FTAPS where one side follows the good curve of Figure 4.1(a), while the other side follows the hot pixel curve of the same figure. The expected illumination response curve is plotted in Figure 4.3 for three idealized cases: a hot FTAPS pixel with minimal, moderate and large dark offset. Dark offset is the *dark current×time* product, which represents that amount of dark current carriers collected throughout a complete exposure. Similarly, illumination energy is used to represent the total accumulated light after a given exposure duration, or equivalently, the accumulated amount of photogenerated charge. From the plot, we see that when the dark current is minimal, the FTAPS responds linearly to illumination until it saturates (corresponding to the curve with minimal dark offset). Equivalently, the same curve results when a very short exposure duration is used, even at non-zero dark current levels because the dark offset is a function of both dark current and exposure duration.

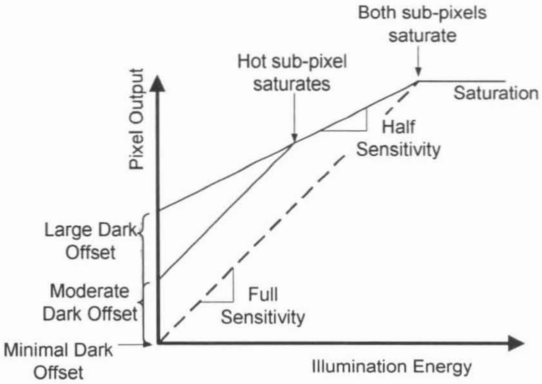


Figure 4.3. Idealized illumination response of an FTAPS pixel with a single hot sub-pixel.

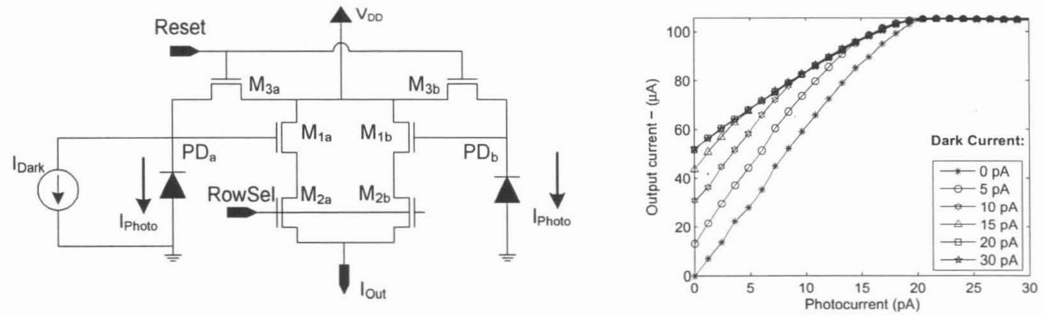
At increased dark currents or exposures (i.e. the moderate dark offset curve), the response curve exhibits an offset and is divided into three operating regions. In the first region, both sub-pixels respond to illumination and the pixel output increases with full sensitivity. Once the hot sub-pixel saturates, only the good sub-pixel can collect more illumination, and the combined output responds at half sensitivity, saturating when both sub-pixels saturate. As exposure duration or dark current increase, the larger dark offset causes the full-to-half-sensitivity transition point to shift further left on the plot.

At long exposure durations or very hot pixels with excessive dark current, any illumination level saturates the hot sub-pixel and no useful information is recorded. In such cases, standard APS designs would generate no useful output, but the working sub-pixel of the FTAPS continues to respond at half the standard sensitivity. Thus, the FTAPS reaches full saturation at the same illumination as an undamaged pixel.

4.2 Circuit Simulation

To validate the predicted FTAPS behaviour with hot pixel defects, HSPICE simulations were performed using the circuit model shown in Figure 4.4(a), where the dark current is modelled as an additional current in parallel with the photocurrent. Figure 4.4(b) shows the resulting illumination response. All simulations were conducted with a short, 2ms, exposure duration, but are representative of results at any duration.

Simulated as a working pixel without dark current (i.e. 0 pA), the FTAPS responds linearly to illumination at full sensitivity until it saturates. At moderate dark current levels, up to 5-10 pA in Figure 4.4(b), the pixel initially responds with full sensitivity and transitions to the half-sensitivity region when the hot sub-pixel saturates, matching the "moderate dark offset" curve predicted in Figure 4.3. In extreme cases, when dark current approaches 30 pA, the hot sub-pixel saturates regardless of the illumination intensity and the pixel responds with half of its nominal sensitivity throughout the entire illumination range. Again, these simulated curves match well with the predicted "excessive dark offset" response in Figure 4.3. Thus, the simulated model of a hot FTAPS matches closely with the expected response characteristics, albeit not as linearly. It should be noted that even standard, commercially deployed pixel designs exhibit measurable non-linearity. The final linear response is in fact a result of signal processing applied after each exposure (i.e. the linearization step discussed earlier).



(a) Simulation circuit model. *(b)Simulated illumination response.*

Figure 4.4. HSPICE circuit simulation of FTAPS with hot pixel defect. (a) Simulation circuit model. (b) Simulated illumination response.

4.3 Recovery Algorithm

The true (i.e. defect-free) output of an FTAPS pixel can be recovered from the defect-affected output by first estimating the magnitude of the dark current impact and subtracting that value in light of the known pixel response. However, the algorithm outlined here is significantly more complex than darkframe subtraction because the non-linear response of the current-mediated FTAPS must be considered. Moreover, the hot sub-pixel typically operates near saturation, which further limits the linear range of operation and makes recovery more challenging. Nonetheless, most camera systems already include dedicated image processing hardware that can be adapted to carry out the limited number of correction operations required for the low fault densities observed thus far.

As a prerequisite to this algorithm, the illumination response of a good sub-pixel is characterized at the factory and stored in memory. The output current, I_{out}, at each hot pixel is then described as the sum of the outputs from a good sub-pixel and a hot sub-pixel,

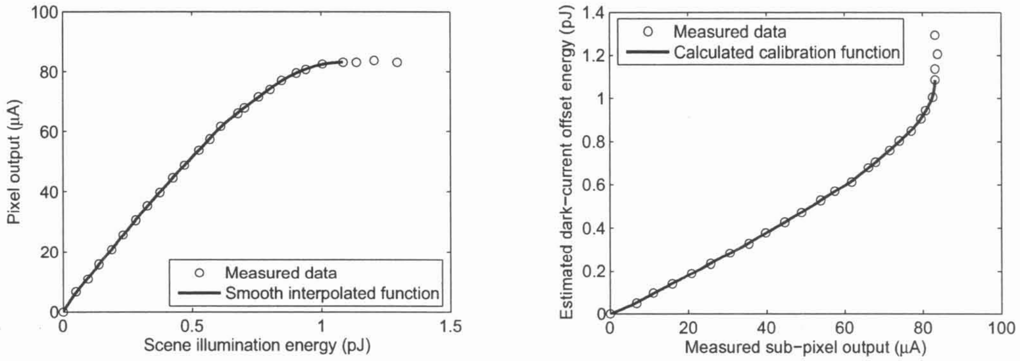
$$I_{out} = f(Q_{photo} + Q_{dark}) + f(Q_{photo}), \qquad (4.1)$$

where Q_{photo} and Q_{dark}, are the charge resulting from integration of photocurrent and dark current, respectively. The function $f(Q)$ is the illumination response characteristic of a single good sub-pixel.

Ideally, the sensor is calibrated at the factory and $f(Q)$ is stored as a common look-up table shared by all pixels across the sensor. The characterization results discussed in Section 3.2 demonstrated that the sensitivity of hot pixels remains the same as that of good pixels, allowing the same calibration curve to be applied to faulty pixels. However, the sensitivity varied considerably from pixel to pixel in the sensor arrays used for testing, necessitating the calculation and storage of individual functions for each tested pixel. Imagers fabricated in carefully monitored, dedicated processes are not expected to show this spread in sensitivity, further simplifying the algorithm.

Figure 4.5(a) and (b) respectively show sample forward and inverse calibration functions, where the markers indicate measured data points and the solid lines shows the smooth f(Q) and

$f^{1}(Q)$ obtained by linear interpolation.. These functions were obtained in the same way they might be recorded in the factory: the illumination response of a complete FTAPS pixel was measured and the result divided to 2 to give the response of a single sub-pixel.



(a) Forward calibration curve, $f(Q)$. *(b) Inverse calibration curve, $f^{1}(I_{out})$.*

Figure 4.5. (a) Forward and (b) inverse calibration curves used to recover data from hot FTAPS pixels.

Once an image is captured, the correction process begins by locating all hot pixels. A darkframe is captured with the same exposure duration, and hot pixels are identified as bright pixels in this image. At each identified hot pixel, we record the pixel values from the darkframe, denoted as $I_{darkfrome}$, and the photograph, denoted as $I_{picture}$. The accumulated charge due to dark current, Q_{dark}, is then estimated from the darkframe value using the reverse calibration curve. Given this dark offset and forward calibration curve, an expected illumination response curve is constructed, and the photogenerated charge in the good pixel, Q_{photo}, is determined by minimizing the error between the expected curve and the pixel value in the final image.

Summarized from the start, the correction process proceeds as follows:

1. Build the illumination response look-up function, $f(Q)$.
2. From the captured image record the output, $I_{picture} = f(Q_{dark} + Q_{photo}) + f(Q_{photo})$.
3. From the darkframe, record the output $I_{darkframe} = f(Q_{darkframe})$.
4. Estimate $Q_{dark} = f^{-1}(I_{darkframe})$.
5. Estimate Q_{photo} such that $I_{picture} \approx f(Q_{dark} + Q_{photo}) + f(Q_{photo})$ is satisfied.

6. Map Q_{pohto} onto the range used by the image format and store to the image file.

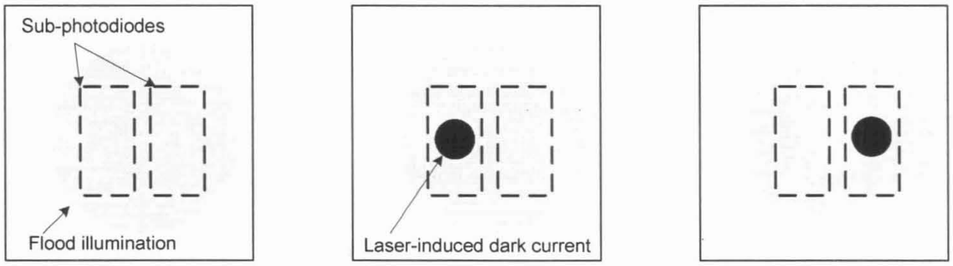
The correction accuracy of this algorithm will now be experimentally validated in the following section.

4.4 Experimental Configuration

The efficacy of this hot pixel recovery technique was evaluated experimentally using FTAPS arrays fabricated in standard CMOS processes. However, creating hot pixels by inflicting displacement damage using radiation sources would be impractical because the magnitude and distribution of faults would be difficult to control. In addition, it is challenging to generate sufficient energetic particles to ensure a sufficient quantity of interactions between impinging particles and sensor materials. Instead, hot pixels were emulated by inducing an additional current in parallel with the typical photocurrent in a single sub-pixel (e.g. the equivalent circuit of Figure 4.4 was implemented). Two methods were employed in two stages of testing: an optical method that can be applied to any FTAPS sensor, and an electrical method using specially-designed FTAPS arrays. Details of both test methods, the test sensors, and the control equipment are given in the following sections.

4.4.1 Optical Hot Pixel Emulation

In this optical emulation technique, the "dark current" is induced by creating an additional illumination in half of the pixel. This added light is created by focusing a laser onto a single ordinary FTAPS sub-pixel while a second broadband light source provides flood illumination to the entire array. Figure 4.6 illustrates how the two illumination sources combine to generate the fault conditions. The flood illumination source covers the entire sensor array with a uniform lightfield that represents the scene. The primary benefit of optical emulation is that it can be readily applied to any existing FTAPS for quick evaluation of the technique. However, the complexity of the optical apparatus makes it impossible to automate testing and thus a limited sample set could be acquired.



(a) Good pixel. *(b) Hot left sub-pixel.* *(c) Hot right sub-pixel.*

Figure 4.6. Hot pixels emulated by selective laser illumination of sub-pixels.

In these experiments, an incandescent microscope lamp was used to provide the flood illumination, as it is designed to provide a uniform light field. Likewise, a focused 488 nm Argon ion laser produced the "dark current." Two separate optical power meters were used to monitor and set both illumination levels independently. Moreover, to facilitate comparison between incident illumination energy and the energy predicted by the correction algorithm, the output of the broadband flood illumination source was calibrated to the output of the Argon laser. That is, each specified flood illumination energy corresponds to the laser energy required to produce the same output level from a typical FTAPS pixel.

To emulate faulty pixels of varying hotness, the laser illumination intensity at each pixel was swept across a range of values, inducing larger levels of dark current, while the flood illumination was also swept across a range of values to generate varying levels of scene brightness. Due to the symmetry of the design, all experiments were conducted with the laser aimed at the left sub-pixel only.

4.4.1.1 Test and Control Hardware Implementation

The test chip used with this method was an array of FTAPS pixels implemented in a standard, single-polysilicon, 4-metal, non-silicided 0.35 μm CMOS process by Cory Jung [20].

As shown in Figure 4.7. power, digital and analog control signals to the device under test (DUT) were all generated by a PC running a modified version of LabVIEW software designed by previous graduate students [16, 17]. Off-chip trans-impedance amplifiers converted the FTAPS' current-mode output to voltage signals that were recorded by a digital storage oscilloscope.

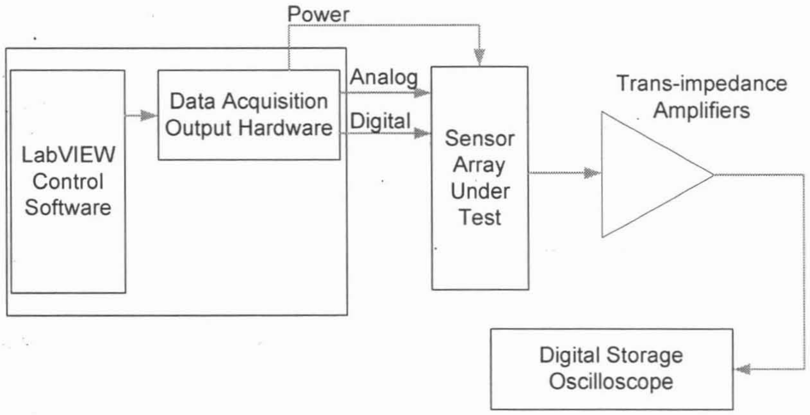

Power
LabVIEW Control Software
Data Acquisition Output Hardware
Analog
Digital
Sensor Array Under Test
Trans-impedance Amplifiers
Digital Storage Oscilloscope

Figure 4.7. Control equipment for optical emulation experiments.

Laser and flood illumination intensity were controlled manually using their respective interfaces. A sub-micrometer precision computer-controlled 2-axis table was used to aim the laser at individual sub-pixels.

4.4.2 Electrical Hot Pixel Emulation

To overcome the limited test sample size of the optical test method, an FTAPS array was designed with dedicated in-pixel transistors that electrically induce dark current. This test circuit, shown in Figure 4.8, allows the emulated dark current to be controlled by an external analog signal, V_{Fault}. Note, however, that I_{Dark} and I_{Photo} are very low magnitude signals, which requires that M_{Test} operate in the sub-threshold region to generate accurate results.

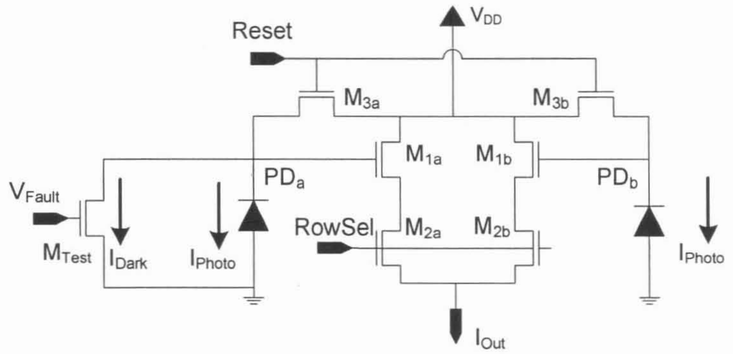


Figure 4.8. Schematic of FTAPS test pixel with electrically-emulated hot pixel fault.

4.4.2.1 Test and Control Hardware Implementation

The test circuit was implemented for this thesis in a standard, single-polysilicon, 6-metal, silicided, 0.18 μm CMOS process tuned for logic circuits. A total of 841 pixels of various designs were included on the 1.2 mm × 1.4 mm chip.

Similar to the procedure used for the optical emulation, a LabVIEW software-controlled PC generated power, analog and digital control signals sent to the test sensor, including the dark current control signal, V_{Fault}. (see Figure 4.9). Off-chip trans-impedance amplifiers again converted output currents to voltage signals, which were measured and recorded directly by the data acquisition (DAQ) PC.

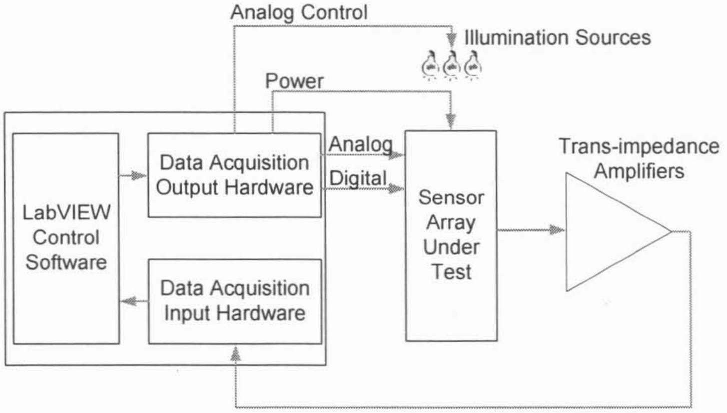


Figure 4.9. Control equipment for electrical emulation experiments.

Flood illumination was generated by an array of current-controlled red LEDs, which were commanded by the same DAQ equipment controlling the sensor. A glass diffuser plate created a uniform lightfield with less than 5% nonuniformity at the sensor array.

4.4.2.2 Limitations

Unlike the 0.35 μm sensors used previously, the 0.18 μm sensors exhibited very large dark current, which compressed the dynamic range and introduced anomalies into the response of test pixels. This increased dark current can be attributed trap states created in or near the photodiode by two key features of the fabrication process: metals used in the silicide process can travel into the Silicon bulk and create trap states; and shallow trench isolation (STI) creates many surface traps at the Si-SiO_2 interface [61].

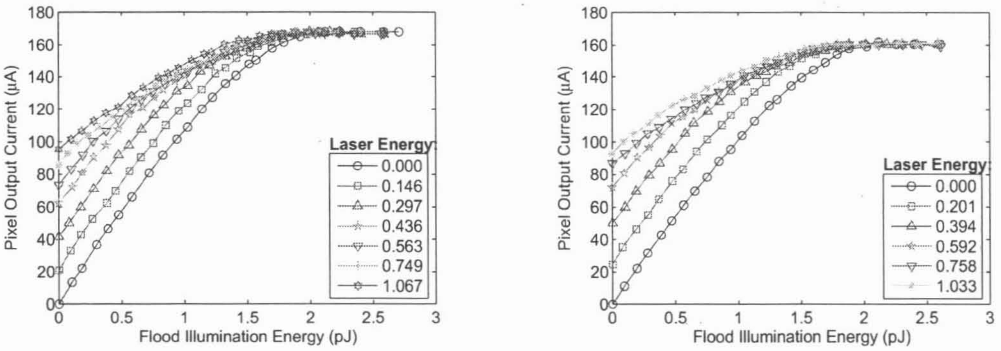
To minimize the impact of these uncontrollable dark currents, experiments with the 0.18 μm sensors were conducted at a very short exposure duration: 4 ms. Consequently, very intense flood illumination and high V_{Fault}-induced dark current were required to exercise the complete dynamic range of test pixels. These large dark and photo currents competed with the current through M_{3a} (see Figure 4.8) during pixel reset, which in turn created a voltage drop across M_{3a} and prevented the photodiode from being complete drained of electrons. Consequently, the pixels' internal reset states varied by as much as 22% between best and worst-case conditions. In most cases, this reset level reduction was calibrated out of the results, but some errors remain at the worst-case end of the spectrum, particularly because of the FTAPS' highly non-linear response.

4.5 Characterization Results

The following sections describe experimental results confirming the viability of hot pixel mitigation using the FTAPS and an appropriate correction algorithm. In both experiments, the aim of correction is to recover the illumination intensity impinging on the hot pixel using the faulty output data. Correction accuracy is determined by comparing the measured illumination intensity to the recovered value.

4.5.1 Optical Emulation

Figure 4.10 below shows the measured response curves for two representative pixels. Qualitatively, the hot FTAPS pixels behave as predicted, with increasing levels of dark current adding an increasingly-larger offset to the output until the hot sub-pixel becomes saturated. However, the response characteristics are more strongly non-linear than the idealized or simulated curves described in Section 4.1. Some of this non-linearity can be attributed to the quadratic response of the in-pixel transconductance amplifier (M_2 in Figure 4.2) and is accurately predicted by the earlier HSPICE simulations above. However, pixel sensitivity is further reduced at high flood illumination intensities because the photo-collection volume is reduced by the shrinking of the photodiode depletion region as its reverse-bias is discharged.

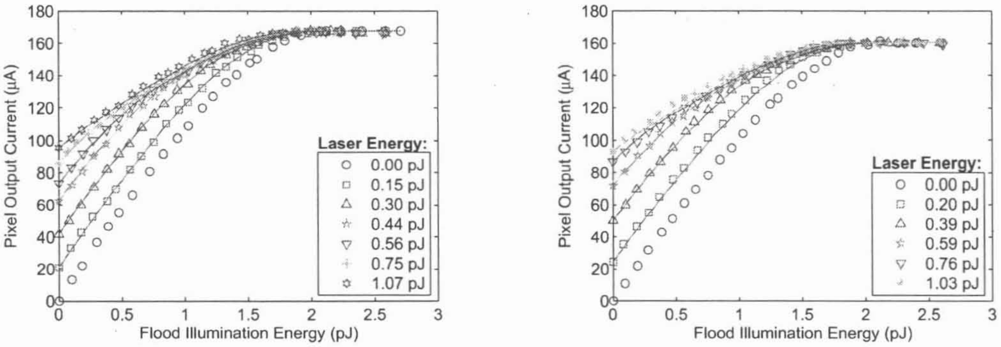


(a) Pixel A. *(b) Pixel B.*

Figure 4.10. Measured illumination response of 2 FTAPS pixels with emulated hot pixel faults.

Thus, hot FTAPS pixels clearly generate usable output data under all hot fault conditions, making this a viable technique for isolating hot defects in the field. The impact of increased non-linearity and other non-ideal characteristics are best quantified by evaluating how accurately the true scene illumination can be recovered from faulty FTAPS output data using the accompanying correction algorithm.

Figure 4.11 shows the pixel response predicted by the correction algorithm (solid lines) overlaid on the measured data points (markers). At a given output current, the horizontal distance between the estimated and actual incident illumination represents the error in the correction algorithm. Thus, the accuracy of the algorithm is measured by its ability to recover the true illumination intensity that was incident on the pixel. At all dark current levels, the predicted curves appear to pass directly through the measured marker points, indicating a high degree of correction accuracy.



(a) Pixel A.

(b) Pixel B.

Figure 4.11. Measured and predicted illumination response curves of two pixels as used by the correction algorithm.

The prediction error has been compiled into a single statistic for all five of the pixels tested at each dark current level and is displayed in the histogram showed in Figure 4.12. To enable seamless comparison between experiments, the estimation error (difference between predicted and actual incident light energy at a given current level) has been normalized to the illumination dynamic range of the pixel under test. Thus, "normalized light energy estimation error," is the estimation error, measured as illumination energy, normalized by the lowest illumination energy that causes the pixel to saturate.

In the vast majority of situations, the correction algorithm is very accurate, recovering the true normalized illumination value within ±0.05 error 96% of the time.

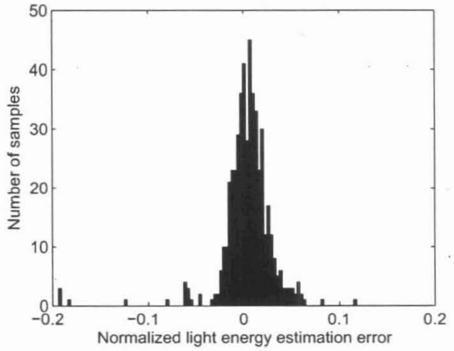


Figure 4.12. Error incurred by hot pixel correction algorithm across all tested pixels.

All of the outliers occurred when both laser and flood illumination were set to very high levels, causing excess carriers to bloom from that saturated hot sub-pixel to the still-working one. In more detail, blooming occurs when the hot sub-pixel photodiode has been largely discharged, minimizing the size of the depletion region and leaving the excess flood illumination-induced carriers free to diffuse to the working sub-pixel, where they are collected. Thus, the effective photosensitive area of the good sub-pixel is increased and with it the effective sensitivity of the FTAPS. Consequently, the correction algorithm underestimates the expected output value under these situations, as highlighted in Figure 4.13. In particular, note how the predicted curve at high laser energy is well below the measured data points, while the error is smaller and randomly distributed at lower laser energy (i.e. dark current).

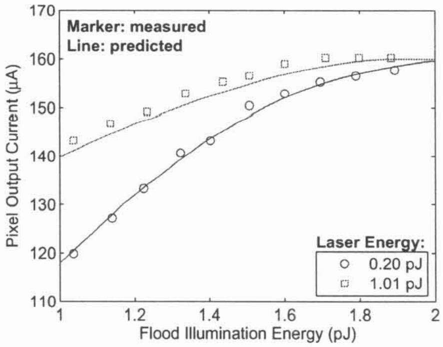


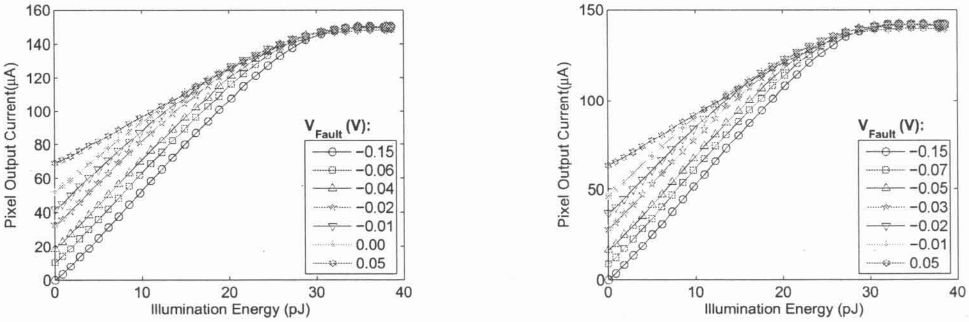
Figure 4.13. Close-up view of hot pixel recovery error from an FTAPS pixel under intense flood and laser illumination.

Correcting for this estimation error would be very challenging to accomplish in software without extensive calibration, making it unlikely that this sensor design could eliminate the worst-case error plotted above. Nonetheless, in 96% of test cases, the recovery algorithm produced excellent results that surpass the dynamic range-limiting correction offered by darkframe subtraction. Furthermore, commercial sensor designs often limit blooming by lowering substrate resistivity or designing dedicated anti-blooming structures into the pixels, both of which may be explored in future sensor designs. At the same time, we must recognize that this hot pixel emulation method does not perfectly model the true defect scenario. In these experiments, both photo and dark current sources are generated by light sources that cover a large area of the photodiode. Thus, both sources inject significant quantities of excess carriers at the periphery of the hot photodiode, allowing those carriers to very easily diffuse to the working half of our design. Conversely, defect-induced dark carriers are most likely created at a point source (see Section 3.3.2), which implies that those carriers must travel longer distances to reach the working photodiode, furthering reducing the likelihood that the dark carriers themselves will contribute to blooming. Photogenerated carriers will remain free to diffuse in the same manner in emulated or defect-induced hot pixels.

4.5.2 Electrical Emulation

A second set of experiments were conducted using the electrical emulation technique in order to gain a larger sample or results and more precise measurements. The raw illumination response curves of two FTAPS pixels with electrically-emulated hot faults are shown in Figure 4.14. The plots describe dark current amplitude by the gate voltage (V_{Fault}) applied to the control transistor. Because these control transistors are operating deep in the subthreshold region, variations between pixels can be quite large and the exact dark current magnitude cannot be precisely predicted. For clarity, only a handful of V_{Fault} levels are displayed, although measurements were collected at 25 distinct dark current amplitudes. In addition, all data points have been corrected as much as possible to account for the reset-level issue discussed earlier.

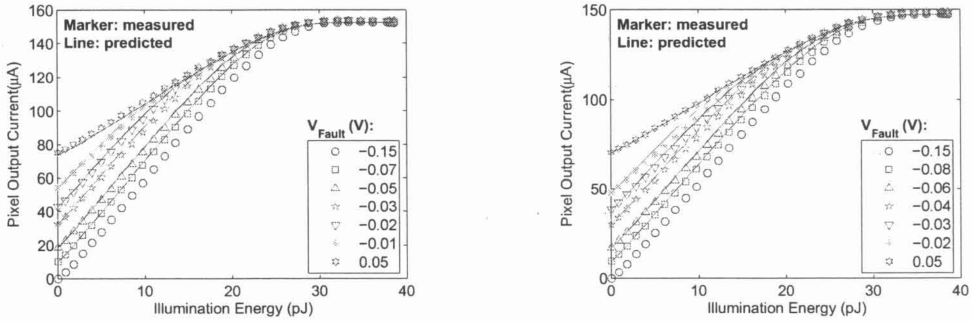
Qualitative analysis of the plots once again suggests that the emulated hot 0.18 µm CMOS sensors behave according to the ideal behaviour outlined earlier in Figure 4.3. When dark current is made very small (e.g. V_{Fault} = -0.15 V), the FTAPS responds linearly until the illumination energy approaches about 25 pJ, where it begins to roll off toward saturation. Between V_{Fault} levels of -0.08 V and about 0 V, the FTAPS is operating in the "moderate dark offset" regime, responding with full sensitivity until one sub-pixel saturates, and subsequently responding at half sensitivity until total saturation. The large dark current regime is represented by the 0.05 V curves in both plots, whereby one sub-pixel is saturated even at 0 illumination energy, and the pixel correspondingly responds at half sensitivity until saturation. In all cases, the response closely follows the idealized curve presented above but with greater non-linearity near the saturation regions.



(a) Pixel A. *(b) Pixel B.*

Figure 4.14. Measured and illumination response curves of two FTAPS pixels with electrically-emulated hot pixel faults.

Figure 4.15 shows the predicted illumination response curve from the correction algorithm (solid lines) superimposed on measured illumination response data points (markers). Unlike the 0.35 µm CMOS sensors tested previously, blooming does not appear to be an issue at intense illumination levels, as evidenced by the close match between measured data points and the predicted curve.



(a) Pixel A. *(b) Pixel B.*

Figure 4.15. Comparison of measured and predicted illumination response curves from two FTAPS pixels with electrically-emulated hot pixel faults.

Once again, the correction algorithm accuracy was taken as the horizontal distance (measured as energy) between predicted illumination energy and actual incident illumination energy. For convenience, this value was normalized to the dynamic range (i.e. smallest saturating illumination energy) of each pixel. The sensor-wide prediction error, collected from 17 FTAPS pixels with electrically-induced faults, has been combined into a single statistic for which the histogram is shown Figure 4.16. In the majority of cases, the normalized error using this more robust experimental method is less than 0.05 and is much better than predicted by the optical emulation method. In fact, 98% of the tested data points yielded a normalized error within ±0.05 (light energy estimation error normalized to pixel's illumination dynamic range).

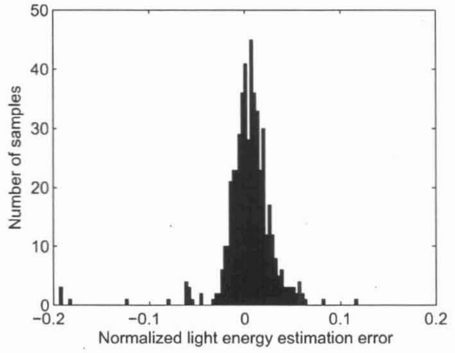


Figure 4.16. Error incurred by hot pixel correction algorithm across all tested pixels.

However, when both dark current and illumination level are extreme, experimental error is greatly increased. Because of the very short exposure times used, the induced dark current must be very large, which makes the reset operation incomplete and reduces the dynamic range of the hot sub-pixel, in turn reducing its apparent sensitivity. Nonetheless, this only occurs at very extreme cases and is an aberration of the experimental method rather than a shortcoming of the fault tolerant design. Dark current magnitudes encountered in the field would typically be an order of magnitude smaller than those induced in this test. Moreover, the correction algorithm could be extended to test for rare situations when hot pixels are operating in this extreme regime and alternate corrective steps, such as replacing the hot pixel with an average of its neighbours, could be used instead of the error-prone value given above.

In summary, two experimental approaches have been demonstrated to validate the hot FTAPS correction technique. Both sets of results closely match the simulated and idealized illumination response behaviour, featuring operating regions with minimal, moderate, and large dark offsets, which consequently cause a 0.5× shift in the FTAPS sensitivity when the hot sub-pixel saturates. In both cases, a comparison of predicted illumination energy with measured incident illumination intensity showed that the true signal value can very accurately be extracted from the faulty FTAPS output.

4.6 Summary

While software correction methods, like darkframe subtraction, can sometimes remove bright spots in images caused by hot pixel faults, they cannot recover the original dynamic range of the device. A fault tolerant pixel architecture was proposed to remedy that limitation by partitioning pixels into two identical halves. Thus, hot defects are isolated to one sub-pixel that may saturate early from increased dark current, while the other sub-pixel continues to operate unaffected. The current-mode output signals from the two sub-pixels are summed at the pixel level, allowing software algorithms to separate the usable signal from the hot offset. Because the sub-pixels share the same control and output signals, the added cost of this redundancy is a minimal reduction in fill factor.

The concept was evaluated experimentally using two techniques to emulate hot pixels in fabricated FTAPS arrays. In the first, optical method, dark current was emulated by preferentially illuminating one sub-pixel with a focused laser beam. In the second technique, a test transistor was included in the pixel design to leak charge from the photodiode. Blooming effects in the sensors used with the optical tests limited the correction accuracy to ±0.05 normalized error in 96% of cases. The more robust electrical tests recovered the true scene illumination within ±0.05 normalized error in 98% of test cases. Extreme photo and dark currents prevented better accuracy in the electrical tests, which could be improved in future designs by limiting intrinsic dark current.

CHAPTER 5

ON-LINE FAULT IDENTIFICATION ALGORITHM

Although not seen in the characterization experiments of Chapter 3, many of the complex fault types listed in Table 2.2 have been reported by users and do threaten to degrade sensor quality over time. Such faults can be overcome by identifying their location and type and correcting for them in-field using software algorithms, but we saw in Chapter 3 that calibrating a sensor is a challenging process. While darkframe data is relatively easy to obtain, generating a uniform lightfield requires relatively complex hardware and optics. Automatically generating a uniform illumination to self-calibrate a camera in the field is nearly impossible without greatly increasing system cost and complexity. Thus, a new algorithm has been developed to use ordinary photographs as inputs for an on-line defect identification system.

The initial motivation for this algorithm was to create a general-purpose self-healing sensor by detecting fully stuck faults on the fly in an FTAPS imager. However, for a self-repairing system like the FTAPS to be useful, it must be able detect the faults in the field rather than relying on cumbersome laboratory calibration. Thus, an FTAPS pixel with one half stuck low could be readily identified in the field and corrected simply by multiplying the output by 2. Using an intelligent, robust algorithm, the entire procedure could occur without human intervention or knowledge.

At the same time, the detection system is also valuable when manufacture-time defects, including those that lead to the fault types listed in Table 2.2, are considered. Factory calibration would typically be used to locate these faults and mask them in software. However, because the calibration process is time-consuming and can significantly add to the cost of a sensor, a cheaper solution is to transfer the calibration burden to the final application. Consider, for example, the small and defect-prone cell phone cameras that must sell for less than a few dollars to be commercially viable. In-field calibration of these sensors would noticeably reduce production costs as well as add in-field robustness.

Moreover, a simple, automated detection algorithm could be used to extend even further the in-field characterization study from Chapter 3. By eliminating any specialized, labour-intensive calibration procedures, a very large sample of digital cameras could be examined easily and simply with minimal user interaction. The resulting data would potentially yield both spatial and temporal fault information. In fact, the wealth of metadata stored in standard digital images (e.g. capture date and time, all camera settings, lens used, etc.) enables photograph analysis to collect as much information as a more cumbersome, hands-on laboratory experiment. Therefore, we would benefit greatly from the development of a robust fault identification algorithm capable of locating a large range of fault types. Such a system should be free of the errors that plague the existing identification methods discussed later in this chapter.

The algorithm presented here goes beyond early fault detection systems by identifying both the type and location of a large class of faults with perfect accuracy. Furthermore, only regular images are required as input, enabling this system to be applied to existing cameras without any sensor hardware modifications. For example, sequences of images may be downloaded from cameras to a host computer for automated analysis and the images could be corrected accordingly. Future cameras could also implement this fault tolerance without additional hardware cost because the integrated firmware could be used to detect faults online and correct them on the fly without the need for a host computer.

This chapter begins with a brief review of existing fault detection methods, followed by a description of how the sensor and faults are modelled by this fault identification algorithm. Three distinct algorithms are then described in detail, with each algorithm increasing in complexity and building on the others' shortcomings. Results from Monte-Carlo simulations of proof-of-concept implementations of each algorithm are then presented, demonstrating the effectiveness of this identification scheme in mitigating the effects of specific fault types.

5.1 Comparison to Existing Methods

Traditionally, imager faults have been corrected by performing factory lightfield calibration to map fault locations and replacing the image pixels with a corrected interpolated value from neighbouring pixels. However, this simple method has three major shortcomings. First, many faults, such as abnormal sensitivity faults, still contain useful information that is simply discarded by blind interpolation. Although some algorithms have been proposed to intelligently correct some faulty pixels by, for example, multiplying the output of a half-sensitivity pixel by two, these systems also rely on factory calibration to generate a fault map[62]. The second drawback is that factory calibration is expensive. Illuminating and reading sensors to calculate and update a fault map is a time-consuming step in fabrication that adds to the cost of the final product. Finally, factory calibration produces a static fault map that remains the same throughout the sensor's lifetime. Therefore, this one time calibration precludes the detection of in-field faults that develop after manufacturing unless the camera is returned to the manufacturer for recalibration. In applications like remote sensing or space missions, recalibration is simply impossible, while in other products, like consumer cameras, recalibration is an undesired hassle. The algorithm described in the remainder of this chapter is designed to overcome these limitations by using ordinary photographs to update the sensor fault map in the field.

Some competing algorithms for locating fully stuck and abnormal sensitivity faults during operation have previously appeared in the literature. Jin et al.[63] compare the difference between a pixel and the average of the eight nearest neighbour pixels to a threshold level to determine which pixels are faulty. Tan and Acharya[64] use the minimum difference between a pixel and its neighbours, combined with sequential probability ratio testing on multiple images, to find defects. Both methods tend to identify most defective pixels, but they are also very susceptible to falsely identifying good pixels as faulty (i.e. creating false positive results). Simulations of these techniques often report as many or more false positives than correctly identified faults, which is undesirable because correcting a working pixel also discards useful image information. Furthermore, neither algorithm provides precise details like sensitivity and offset of the faulty pixel, making truly intelligent correction impossible.

5.2 System Model

Throughout this work, an imager is described as an array of $I \times J$ pixels, with the incident illumination at location (i,j) denoted by $x_{i,j}$. For the majority of these initial experiments, only monochrome greyscale sensors are considered. The extension to colour sensors can be made by treating each individual colour as a separate monochromatic plane.

Following the notation laid out in Section 2.4, each pixel produces a normalized output, $y_{i,j}$, which depends on the incident illumination $x_{i,j}$ according to a linear function $y_{i,j} = m\,x_{i,j} + \Delta$, where the sensitivity, m, and offset, Δ, depend on the fault type. To simplify notation, each fault type is referred to by an ordered pair, $F(m,\Delta)$, consisting of its sensitivity, m, and offset, Δ. Fault $F(m,\varDelta)$ is assumed to occur with probability $p_{(m,\Delta)}$.

Consistent with the findings of the in-field defect characterization experiments, faults are assumed to be uniformly distributed across the sensor in relatively small quantities. Without evidence to the contrary from the experiments in Chapter 3, faults are assumed to be caused by spot defects, and neither clustered nor row or column faults are considered in this preliminary work. The proportion of pixels that are faulty is termed the *fault density*.

5.2.1 Fault Types

To detect faults, we need to model how they behave, although sensors of differing designs and technologies and operating in different environments will naturally develop different types of faults. Several fault models, which are collections of possible fault types, are now presented in order of increasing complexity. An identification algorithm for a particular application may implement a given model based on the faults known to occur in that environment or simply based on the faults that have a sufficient impact on image quality.

The *Simple* fault model includes only fully stuck faults that produce pure black or white outputs and might by considered by an inexpensive camera to remove only the most noticeable blemishes. Table 5.1 gives a summary of the fault types considered, their designation, and the transfer equation between normalized input illumination and the pixel output value.

Table 5.1. *Simple* fault model.

Fault type	*Designation*	*Transfer equation*
Stuck high	*F(0, 1)*	$y = 1$
Stuck low	*F(0, 0)*	$y = 0$

The *FTAPS Stuck* model includes more complex fault behaviour that arises when stuck faults individually afflict the sub-pixels of the FTAPS array. Table 5.2 lists all of the possible fault conditions and the condition of each sub-pixel that generates a given fault. Because each sub-pixel drives half of the total pixel's output, a number of "half" faults are created. For example, one working sub-pixel combined with a stuck low sub-pixel creates a pixel with exactly half sensitivity. Although spot defects are unlikely to strike both sub-pixels simultaneously, and two defects are unlikely to simultaneously strike adjacent sub-pixels, double faults are still included for completeness.

Table 5.2. *FTAPS Stuck* fault model.

			Sub-pixel condition	
Fault type	*Designation*	*Transfer equation*	*Left*	*Right*
Fully stuck high	*F(0, 1)*	y = 1	Stuck high	Stuck high
Fully stuck low	*F(0, 0)*	y = 0	Stuck low	Stuck low
Fully stuck mid	*F(0, 0.5)*	y = 0.5	Stuck high	Stuck low
Half stuck high	*F(0.5, 0.5)*	y = 0.5x + 0.5	Good	Stuck high
Half stuck low	*F(0.5, 0)*	y = 0.5x	Good	Stuck low

It is instructive to consider in more detail the relative probabilities that a single FTAPS pixel will become afflicted by any of the faults described above. Here, we treat a fault in either sub-pixel as an independent event. Then, we can denote the probabilities of observing a stuck high or stuck low sub-pixel by p_H and p_L, respectively. The probability of observing a good sub-pixel is denoted by p_G. Table 5.3 shows the resulting expressions for observing any FTAPS fault type given these probabilities. For the sake of example, we can then assume that stuck high and low sub-pixels are equiprobable and occur at the rate of 100 faults per 1 million pixels (note that this fault density is much larger than observed in any sensors tested in this thesis but is used here only for illustrative purposed). Consequently, the probability of observing a good sub-pixel is 0.9998, and the probabilities of all FTAPS fault types are computed in *Example value* column of Table 5.3. We can readily see that fully stuck pixels of any form will indeed be unlikely to develop in any sensor. At these rates, only 1 in every 10 cameras (with 10 million pixels) would develop a fully stuck high FTAPS pixel.

Table 5.3. Relative probabilities of FTAPS faults.

	Sub-pixel condition		*Probabilities*	
Fault type	***Left***	***Right***	***Expression***	***Example value***
Fully stuck high	Stuck high	Stuck high	p_H^2	1×10^{-8}
Fully stuck low	Stuck low	Stuck low	p_L^2	1×10^{-8}
Fully stuck mid	Stuck high	Stuck low	$2\ (p_H \times p_L)$	2×10^{-8}
Half stuck high	Good	Stuck high	$2\ (p_H \times p_G)$	1.9996×10^{-4}
Half stuck low	Good	Stuck low	$2\ (p_L \times p_G)$	1.9996×10^{-4}

The previous defect models have only included abnormal sensitivity faults at specific discrete sensitivity levels, which were represented by a single fault type. However, in-field sensitivity shifts or manufacture-time defects are likely to inject faults with a continuous range of sensitivities such that pixels sensitivities may range anywhere from 0.0 to 1.0. Perfectly identifying the sensitivity of each fault would lead to an infinitely large defect model. Instead, the complete range of sensitivities is partitioned into sub-intervals, where each interval is represented by a fault with some nominal sensitivity. Thus, an interval of physically-possible faults are grouped and diagnosed as a single fault type. The granularity of this grouping offers a tradeoff between diagnostic accuracy and algorithm complexity. For example, an interval of fault types may include real-valued sensitivities anywhere in the range 0.80–0.90, but they would all be assigned the fault type *F(0.85, 0)*. Similarly, faults in the range 0.90–1.0 would be detected as type *F(0.95, 0)*. Therefore, a continuous range of defects is sorted into a finite number of discrete bins to reduce computational requirements. The interval between the two closest fault types in the model is called the *bin size*.

In this preliminary work, two fault models with differing ranges of abnormal sensitivity faults are considered: the *Continuous Half* model includes pixels with sensitivities in the range 0.50–1.0, and *Continuous Full* model extends the range of sensitivities to 0–1.0. As described above, the continuous range of sensitivities in each model is broken down into several equal-width intervals called bins. To avoid confusion, the interval width, called the *bin size*, will be appended to the model name where relevant. Furthermore, the offset, Δ, for all fault types in all three models is assumed zero for simplicity.

For example, the *Continuous Half – 0.10* model includes five fault types, representing pixels with abnormal sensitivity in the range 0.5–1.0, divided into five bins of width 0.10. Table 5.4 shows the fault types and the associated sensitivity intervals that would be included in this fault model. For comparison, the *Continuous Half – 0.025* model would include 21 fault types, each with interval width 0.025.

Table 5.4. Example of the *Continuous Half – 0.10* fault model.

Fault type	*Designation*	*Included sensitivity range*
Good	*F(1, 0)*	1.0 – ∞
0.95 sensitivity	*F(0.95, 0)*	0.9 – 1.0
0.85 sensitivity	*F(0.85, 0)*	0.8 – 0.9
0.75 sensitivity	*F(0.75, 0)*	0.7 – 0.8
0.65 sensitivity	*F(0.65, 0)*	0.6 – 0.7
0.55 sensitivity	*F(0.55, 0)*	0.5 – 0.6

For reference, Table 5.5 gives a list of the fault models that have been described here along with the nature of fault types included in each model.

Table 5.5. Summary of fault models.

Model name	*Included fault types.*
Simple	Simple fully stuck faults.
FTAPS Stuck	Stuck and discrete partial sensitivity faults occurring in the FTAPS.
Continuous Half	Continuous range of abnormal sensitivities in the interval 0.60–1.0.
Continuous Full	Continuous range of abnormal sensitivities in the interval 0.00–1.0.

5.3 Algorithm

The identification algorithm performs a statistical calibration of the sensor by replacing uniform lightfield calibration frames with a sequence of normal photographs. In analyzing each image, the algorithm calculates the probability that a particular fault type would produce the observed output value. These likelihoods are collected over an entire sequence of images to balance variations in image scene content, and a decision is made based on the overall most probably fault type. For instance, consider the simple example of a stuck low pixel. In a single image, there exists a finite, non-zero probability that this pixel is good and produced a correct black. However, it is unlikely that this pixel should be black in an entire sequence of images, and thus the likelihood of that pixel being good diminishes with each tested image. At the end of the process, our sample pixel is clearly stuck low because that probability is the greatest.

In more detail, a sequence of T photographs is considered in addition to an optional darkfield (considered the 0th image). At a given pixel location (i,j), this results in a sequence of samples $Y = y^{(0)},y^{(1)},\ldots,y^{(T)}$, which is used as evidence in forming Bayesian Inferences into the likelihood of a particular fault residing at that location (where the i,j indices are omitted for brevity). This statement is evaluated for every fault type, F, and is expressed mathematically as,

$$P\left(F\middle|Y=y^{(0)},y^{(1)},\dots,y^{(T)}\right)=\frac{P(Y|F)P(F)}{\sum\limits_{F'=\{all\ F\}}P(Y\mid F')P(F')}. \tag{5.1}$$

The *a priori* probability, *P(F)*, is the initial likelihood of observing fault *F* without any further evidence, and is usually obtained from previous characterization of faults. The conditional term, *P(Y|F)*, represents how well the observed pixel values match the expected output of a given fault type. This probability is a statistical metric computed from the image sequence.

In practice, the calculations are implemented recursively such that each image in the sequence is processed individually, with previous iterations (images in the sequence) forming the *a priori* probability for subsequent steps. The following equation gives the likelihood of fault *F* after testing the k^{th} image in the series:

$$P\left(F\middle|y^{(0)},y^{(1)},\dots,y^{(k)}\right)=\frac{P\left(y^{(k)}|F\right)P\left(F\mid y^{(0)},y^{(1)},\dots,y^{(k-1)}\right)}{\sum\limits_{F'=\{all\ F\}}P\left(y^{(k)}\mid F'\right)P\left(F'\mid y^{(0)},y^{(1)},\dots,y^{(k-1)}\right)}. \tag{5.2}$$

Thus, after analyzing the k$^{\text{th}}$ image, the system only needs to store $P(F|\ y....y^{(k)})$ for each fault type at each pixel, rather than the individual conditional probabilities. Note that a complete fault model also includes good pixels, such that $P(y^{(k)}\mid F(1.0,\ 0))$ is also evaluated as part of the above sequence. The algorithm proceeds by evaluating Eq. (5.2) for the entire sequence of *T* (or *T+1* if a darkfield is used) images, after which the fault type with the highest likelihood is taken as correct. In this design, the parameter *T* is fixed before testing begins, although more elaborate stopping criteria are possible.

Several algorithms have been considered for implementing the decision making term $P(y^{(k)}|F)$. The simplest approach looks at the image histogram to determine if a pixel value is likely to occur. A second technique estimates the expected value of a pixel by interpolating its neighbours, and uses that result to find the fault type that best fits the discrepancy between expected and observed values. Finally, the robust round robin technique builds on the interpolation statistics method by considering each of the neighbours separately. These algorithms were developed sequentially, with each increasingly-complex implementation overcoming the limitations of the previous iteration and allowing a more complex fault model to be identified. This progression is justified by the Monte Carlo simulation results presented in Section 5.5.

5.3.1 Image Statistics Method

The image statistics method is the simplest implementation because it uses information that is already calculated by most DSCs during image capture. Recall that Eq. (5.2) is evaluated separately at each pixel for every fault type in the model. The conditional probability for fully stuck faults is calculated from Eq. (5.3), which states mathematically that the output from stuck pixels is expected to be constant in all images.

$$P\left(y^{(k)} \middle| F(0,\Delta)\right) = \begin{cases} 1, & y^{(k)} = \Delta \\ 0, & otherwise \end{cases}. \tag{5.3}$$

For example, a pixel cannot have a stuck low fault if its output reaches a non-zero value. In reality, stuck pixel values are never fixed because of the various noise sources in a sensor. Therefore, the unit impulse probability density function (PDF) given in Eq. (5.3) is actually replaced by a function representing the noise distribution. The implementation of this algorithm will use a Gaussian distribution with a mean of Δ and the appropriate variance to represent noise power. Nonetheless, equations showing stuck faults will continue to use the unit impulse to simplify notation.

To arrive at $P(y^{(k)} \mid F)$ for light sensitive faults, we evaluate the possibility of generating that pixel value from the range of illumination intensities striking the sensor. Although we cannot know the true illumination at any one pixel, the image intensity histogram gives a good estimate of the illumination values actually impinging on the sensor. That is, we use the image histogram probability density, $p_Y(y)$, to accurately estimate $p_X(x)$, because the number of faulty pixels is relatively small. Figure 5.1 shows a histogram from a typical image, where many pixels take on mid-level grey values but few are found toward either extreme.

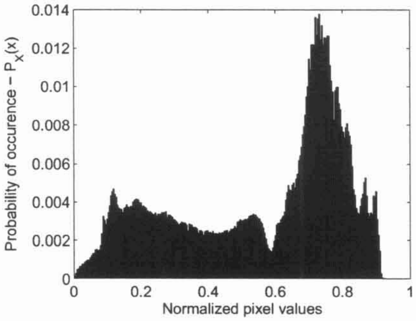


Figure 5.1. Sample image intensity histogram showing probability density $p_Y(y)$.

Now consider the test for a good pixel type at a single location: $P(y_{i,j}^{(k)} \mid F(1,0)) = p_Y(y_{i,j}^{(k)})$. At each image, we determine if that $y^{(k)}$ is likely to occur given the scene content. Any given pixel may record extreme values in some photos, but if the recorded values are consistently improbable then the accumulated likelihood will be small for the good type. Complex faults are now tested by finding the probability of generating the observed $y^{(k)}$ from the combined gain and offset of the fault type given the input illumination, as per Eq. (5.4).

$$P\left(y^{(k)} \middle| F(m,\Delta)\right) = p_Y\left(\frac{y^{(k)} - \Delta}{m}\right), \quad m \neq 0. \tag{5.4}$$

For example, consider the likelihood of a low sensitivity fault (say, a gain of 0.5 or lower) when the observed pixel value is 0.45 in the image matching the histogram of Figure 5.1. To arrive at this middle-grey output, the intensity hitting this test pixel must have been very bright, but the histogram indicates that there were no bright regions in this image. Therefore, *$P(y^{(k)}|F)$* is very small or zero for this fault type at this pixel location.

To better demonstrate how this algorithm would be implemented, Table 5.6 shows a complete list of the equations evaluated at each pixel for the *FTAPS Stuck* fault model. A flowchart of the algorithm is also shown in Figure 5.2 on page 106.

Table 5.6. Equations computed by image statistics method for *FTAPS Stuck* model.

Fault type	$P\left(y^{(k)} \middle| F(m,\Delta)\right)$
Good	$p_Y\left(y^{(k)}\right)$
Fully stuck high	$\begin{cases} 1, & y^{(k)} = 1 \\ 0, & otherwise \end{cases}$
Fully stuck low	$\begin{cases} 1, & y^{(k)} = 0 \\ 0, & otherwise \end{cases}$
Fully stuck mid	$\begin{cases} 1, & y^{(k)} = 0.5 \\ 0, & otherwise \end{cases}$
Half stuck high	$p_Y\left(2\left\lfloor y^{(k)} - 0.5\right\rfloor\right)$
Half stuck low	$p_Y\left(2 \cdot y^{(k)}\right)$

Monte Carlo simulations of the Image Statistics algorithm, detailed in Section 5.5.1-5.5.2, found the method to be very effective at identifying simple stuck fault types. However, this algorithm performed more slowly when identifying the complex fault types in the *FTAPS Stuck* model, necessitating the development of an improved algorithm, described next.

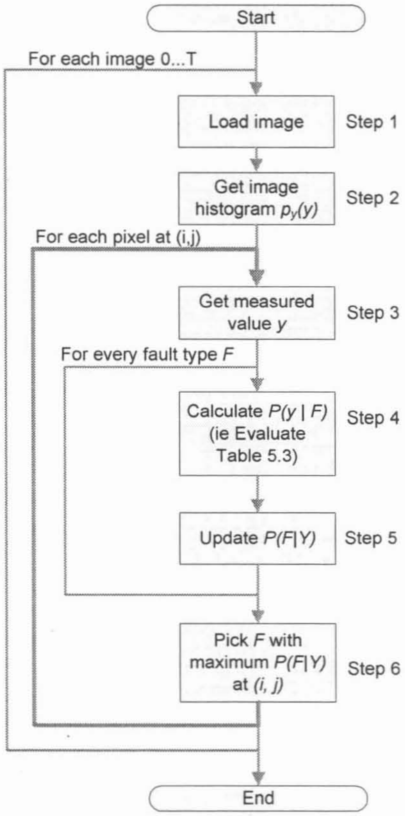


Figure 5.2. Algorithm flowchart for the implementation of the image statistics method.

5.3.2 Interpolation Statistics Method

The image statistics method takes a global approach to estimating the illumination striking a pixel but ignores local information about illumination behaviour, which caused false identifications in simulations and led to long convergence times. For example, a sequence of images with patches of a dark forest near the bottom could cause those pixels to be misidentified as low-sensitivity because of their consistently dark output. To remedy this shortcoming in identifying complex fault types, the interpolation statistics method was devised to utilize information from neighbouring pixels to determine the expected output of a pixel.

The likelihood of fully stuck faults is found using Eq. (5.3) in an identical manner to the image statistics method. Detecting complex fault types begins by assuming the pixels surrounding the test pixel are good and using their outputs to estimate the true incident illumination. As with the image statistics algorithm, we then test if each combination of gain and offset are likely to transform the estimated input into the observed pixel value. If the error between the measured value and transformed estimate is within the accuracy of the estimation method, then a strong match is found. When a dense fault model with many fault types is used, several faults types will initially be equally likely, but processing a long sequence of images will narrow the possibilities to only a single fault type.

In more detail, detection in the k^{th} image starts by making an estimate, $z^{(k)}$, of the incident illumination at the every pixel. The estimation error, $e^{(k)} = z^{(k)} - y^{(k)}$, is then calculated everywhere across image, and the statistics of this error are accumulated in the distribution $p_E(e)$, which describes the effectiveness of the estimation scheme without defects. At every individual pixel, the likelihood for each fault type is determined by computing the error, e_F, between the estimated value and the observed value transformed using the gain and offset for that type. The likelihood of e_F arising from pure interpolation error is then found by retrieving $p_E(e_F)$ as summarized in Eq. (5.5). The effectiveness of this scheme comes from realizing that the estimate may not precisely predict $x^{(k)}$ at each frame, but the error is "averaged out" as more test images are processed.

$$P\left(y^{(k)} \middle| F(m,\Delta)\right) = p_E\left(e_F^{(k)} = z - \frac{y^{(k)} - \Delta}{m}\right), \quad m \neq 0. \tag{5.5}$$

Thus, unlike [63], the interpolation statistics method does not use fixed thresholds to directly compare pixel values to their neighbours. Rather, a model of the expected fault behaviour is included in the comparison to neighbouring values, which allows more complex fault types, like those with abnormal sensitivity, to be considered. The evidence is once again accumulated over many images, thereby accounting for dynamic variations in scene content and avoiding the inherent inaccuracies of blind, fixed-threshold comparisons. Consequently, simulation results presented in Section 5.5.1-5.5.3 show fault identification can be achieved without false positive results, unlike the fixed-threshold results in [63]. Moreover, this method goes beyond [64] by including specific fault models in each comparison (i.e. testing for a specific sensitivity and offset pair), enabling the algorithm to extract and subsequently correct the actual sensitivity and offset of faults. This detailed modelling may also account for the reduced number of false positives observed in simulation results as compared to [64].

Table 5.7 shows some example calculations performed by the interpolation statistics technique for selected fault types.

Table 5.7. Equations computed by interpolation statistics method for selected fault types.

Fault type	$P\left(y^{(k)} \middle\| F(m,\Delta)\right)$
Good	$p_E\left(z - y^{(k)}\right)$
Full stuck high	$\begin{cases} 1, & y^{(k)} = 1 \\ 0, & otherwise \end{cases}$
Half stuck high	$p_E\left(z - \left[2 \cdot y^{(k)} - 1\right]\right)$

In addition, a flowchart of the complete process is shown in Figure 5.3.

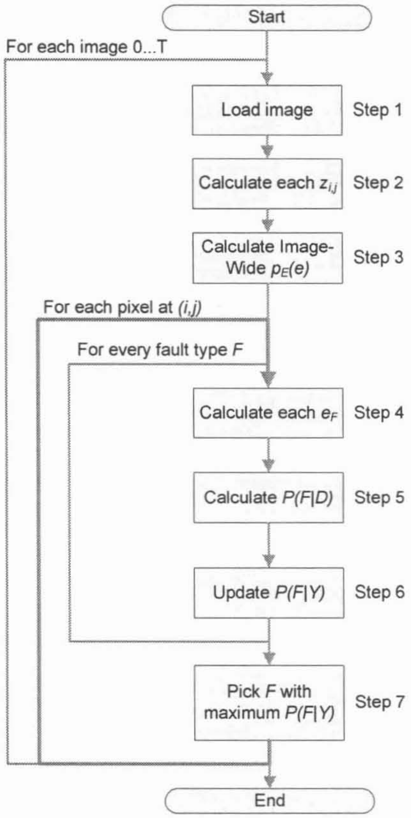


Figure 5.3. Algorithm flowchart for the implementation of the interpolation statistics method.

5.3.2.1 Interpolation Schemes

Although numerous highly-accurate interpolation schemes are available, fault identification places emphasis on different performance parameters than typical applications. Most techniques use many neighbouring pixels to accurately estimate the missing value, but the probability of including multiple faulty pixels in the calculation also increases with the area of interest. Furthermore, most schemes strive to maintain high-frequency image components, while the ideal detection algorithm should smooth the impact of point defects in the image. Finally, computation time is a concern because interpolation may have to be performed many times before a solution is reached. As such, three simple interpolation schemes are considered here: an average of the 4 nearest neighbour (4NN) pixels, an average of the 8 nearest neighbour (8NN) pixels, and a biquadratic method [65] as defined by Eqs.(5.6-5.8), respectively. Note that while all three schemes are described here, the biquadratic method was omitted from experiments due to a clerical error.

$$x_{i,j,4NN} = \tfrac{1}{4}\left(x_{i-1,j} + x_{i,j+1} + x_{i+1,j} + x_{i,j-1}\right). \tag{5.6}$$

$$x_{i,j,8NN} = \tfrac{1}{8}\left(x_{i-1,j} + x_{i-1,j+1} + x_{i,j+1} + x_{i+1,j+1} + x_{i+1,j} + x_{i+1,j-1} + x_{i,j-1} + x_{i-1,j-1}\right). \tag{5.7}$$

$$x_{i,j,Biquad} = \tfrac{1}{2}\left(x_{i-1,j} + x_{i,j+1} + x_{i+1,j} + x_{i,j-1}\right) - \tfrac{1}{4}\left(x_{i-1,j+1} + x_{i+1,j+1} + x_{i+1,j-1} + x_{i-1,j-1}\right). \tag{5.8}$$

Each equation can be implemented by 2-dimensional convolution with the kernel coefficients shown in Figure 5.5.

	$\frac{1}{4}$	
$\frac{1}{4}$	$x_{i,j}$	$\frac{1}{4}$
	$\frac{1}{4}$	

(a) 4NN averaging

$\frac{1}{8}$	$\frac{1}{8}$	$\frac{1}{8}$
$\frac{1}{8}$	$x_{i,j}$	$\frac{1}{8}$
$\frac{1}{8}$	$\frac{1}{8}$	$\frac{1}{8}$

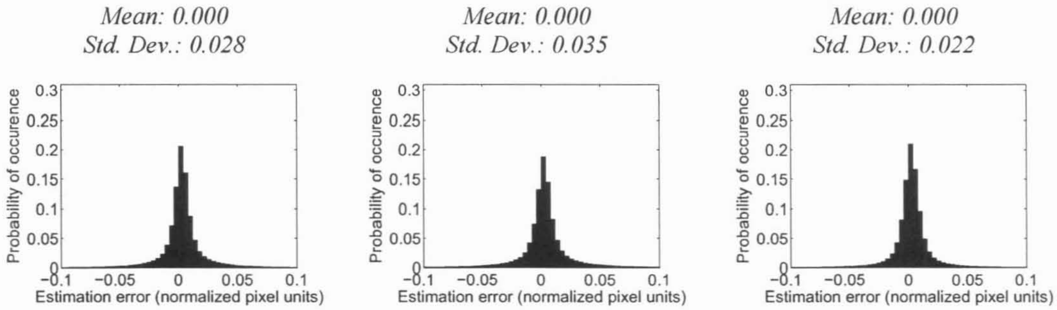
(b) 8NN averaging

$\frac{-1}{4}$	$\frac{1}{2}$	$\frac{-1}{4}$
$\frac{1}{2}$	$x_{i,j}$	$\frac{1}{2}$
$\frac{-1}{4}$	$\frac{1}{2}$	$\frac{-1}{4}$

(c) Biquadratic interpolation

Figure 5.4. Two-dimensional filter kernels for (a) 4NN averaging (b) 8NN averaging and (c) biquadratic interpolation schemes.

To evaluate the accuracy of the interpolation schemes, each method was applied to a bank of defect-free images (see Section 5.4), and the error was calculated between the estimate and actual image value at every pixel. Figure 5.5 shows histograms of the resulting estimation error distribution, $p_E(e)$, obtained for each scheme. Although the graphs show that all three algorithms yielded similar results, 4NN averaging provided the best compromise between typical and worst-case accuracy, which it achieved with the least computation effort.



(a) 4NN averaging.

(b) 8NN averaging.

(c) Biquadratic interpolation.

Figure 5.5. Estimation error in fault-free images using (a) 4NN (b) 8NN and (c) biquadradatic interpolation schemes.

5.3.3 Round Robin Method

Simple interpolation methods occasionally make a very large estimation error when operating near defects or fine image features, thus making it difficult for the interpolation statistics method to distinguish between fault types with small variations in gain. For example, consider the common situation depicted in Figure 5.6, where a black feature edge is located adjacent to our faulty, *F(0.60, 0)* pixel under test (cross-hatched, at centre). Now, when the estimation uses the 4 neighbouring pixels to compute $z_{i,j}$, the result will be drastically lower than the correct value because 25% of the input values are black. Consequently, the fault type with the best fit to the observed pixel response would be *F(0.80, 0)* rather than the correct *F(0.60, 0)*. Ordinarily, the next analyzed image would contain different features and the estimation error would recover to a more reasonable value. However, in the rare, but possible, event that a dark feature edge lands in the same location in multiple images, the incorrect decision can become solidified in the Bayesian decision-making logic, leading to an incorrect result. Such subtle errors will not be noticeable when coarse fault models like *FTAPS Stuck* are employed. However, the highly similar fault types in fine-grained models like the *Continuous* family are more susceptible to these (rare) errors.

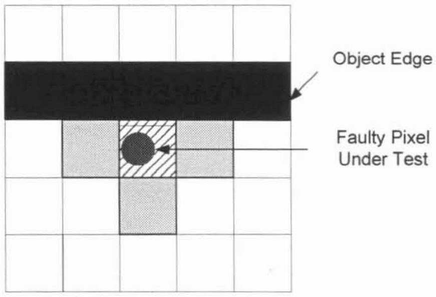
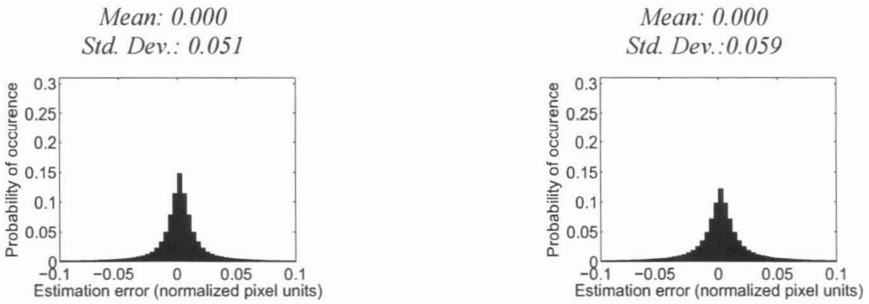


Figure 5.6. Estimation error influenced by neighbouring feature edge.

Therefore, to prevent a single pixel error from influencing the results so strongly, the 4NN interpolation statistics method is partitioned into four separate tests to create a majority-voting scheme called the round robin method.

The algorithm executes as follows: first, pixel *(i, j)* is compared to its neighbour to the left, and the probability $P(F|\, y^{(k)})$ is evaluated; next, the pixel is compared to its neighbour above, and $P(F|\, Y^{(k)})$ is evaluated using the left-pixel test result as its prior probability. The process continues until all four neighbours have been included, and then the $k+1^{th}$ frame in the sequence is examined. This four-test sequence is called the 4 round robin (4RR) test. Likewise, 8NN interpolation statistics can be partitioned to form the 8 round robin (8RR) test.

Figure 5.7 shows histograms of the resulting estimation error when the round robin methods are applied to a bank of fault-free images. As expected, round robin estimation error is greater than the traditional interpolation methods, but the scheme offers the advantage of mitigating the effects of a catastrophic error while maintaining low computational cost.


Mean: 0.000
Std. Dev.: 0.051

Mean: 0.000
Std. Dev.:0.059

(a) 4RR. *(b) 8RR.*

Figure 5.7. Estimation error in fault-free images using (a) 4RR and (b) 8RR methods.

The round robin algorithm was found to be particularly effective at identifying abnormal sensitivity faults; simulation results are presented in Section 5.5.3.

5.4 Monte Carlo Simulation

The effectiveness of each identification algorithm was evaluated with a Monte Carlo software simulator that implements the system model pictured in Figure 5.8. In this model, simulations begin by adding to a sensor randomly placed faults. Then, a scene is created by loading a random image from a bank of known photographs and is transformed according to the fault map of the simulated sensor (e.g. stuck-high pixels become white and the values of half-sensitivity pixels are reduced accordingly). The resulting image, which represents the output from a faulty sensor, is passed to an algorithm from Section 5.3 to locate and identify faults. The simulator then compares the identified fault map to the simulated sensor fault map to measure accuracy, and the cycle is repeated until all *T* pictures have been processed.

Once a complete image sequence has been analyzed, a new faulty sensor is generated and the complete process is repeated for another random sequence. In the simulation experiments described next, all but the most computationally-intensive simulations were repeated 100 times to ensure thorough simulation coverage before varying parameters for the next experiment.

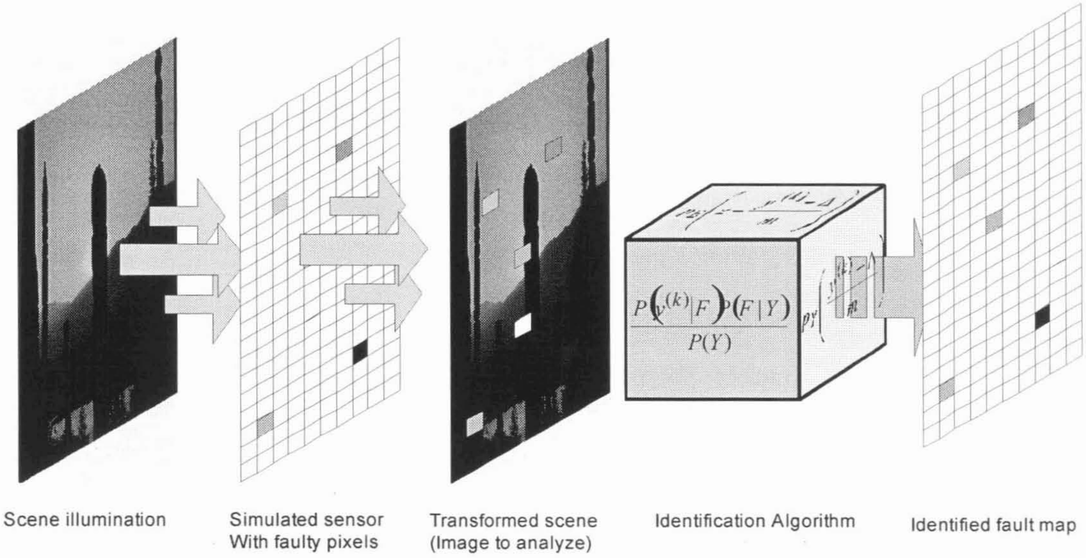


Figure 5.8. Simulation system model with scene, sensor and detection algorithm.

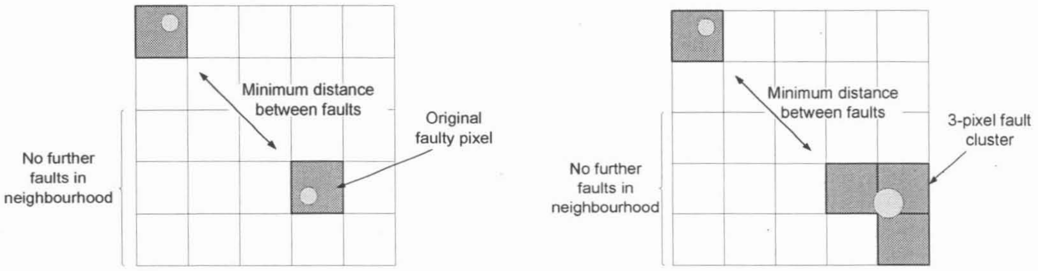
5.4.1 Fault Distribution

In line with the findings of Chapter 3, which showed that faults developed independently without clustering, the simulator creates sensor fault maps by first generating a list of uniformly distributed pixel locations. Next, each location is assigned a fault type, which is straightforward for simple fault types, like those found in the *Simple Stuck* and *FTAPS Stuck* models, because each fault type has a single sensitivity and offset associated with it. To simulate ranged fault types (i.e. from the *Continuous* models), each fault location is assigned a sensitivity value from a uniform random distribution with the appropriate bounds. As such, sensors simulated with a *Continuous* fault model are afflicted by a continuous range of sensitivities.

In every simulation, all fault types are made equiprobable, and the total number of faults in the simulated sensor is $(I \times J) \times p_{total}$, where I and J are the sensor dimensions, and p_{total} is the total fault density given by the sum of the individual fault probabilities. In general, there appeared to be no interaction between different fault types, such that the exact proportion of fault types used in simulation had little bearing on the algorithm's effectiveness.

In most experiments, the uniform random, cluster-free fault distribution was strictly enforced by ensuring that only one fault was assigned to each 3×3 pixel region (i.e. faults were assigned in the 9-pixel pattern shown in Figure 5.9(a)). This step was necessary because uniformly distributed faults can form small clusters with a significant probability when high fault densities are considered. In such cases, algorithm performance can degrade prematurely because multiple incorrect pixel values in a single neighbourhood reduce interpolation accuracy and cause fault identification errors. Therefore, because existing experimental evidence indicates that fault clustering remains highly improbable in most cases, clusters were artificially inhibited to enable other aspects of the identification scheme to be more easily studied.

Nonetheless, clustering was explicitly included in some experiments with the aim of developing a robust detection algorithm capable of handling complex defect models. In these configurations, fault clusters were distributed according to a simplified common-centroid model. That is, a sequence of uniform random locations was again generated, representing the centre of each cluster. Then, each location was allocated as cluster of faults ranging in size from 2 to 8 pixels. Only adjacent-pixel clusters were considered, and the shape of each cluster was fixed according to its size (see Figure 5.9(b) for an example of a 3-pixel cluster fault). Fault types at pixels were assigned independently of other faults in the cluster.


No further faults in neighbourhood
Minimum distance between faults
Original faulty pixel
No further faults in neighbourhood
Minimum distance between faults
3-pixel fault cluster

(a) Typical no clustering case. *(b) Forced clustering.*

Figure 5.9. Fault cluster configurations in simulation experiments.

5.4.2 Performance Metrics

Fault identification effectiveness was measured in one of two ways: speed of convergence or accuracy. When processing the simple stuck fault models, the algorithm was able to fully identify the location and type of all faults without creating any false-positive or false-negative diagnoses. Therefore, performance in these simulations was measured as the minimum number of images required to be analyzed before all faults were found (i.e. the minimum image sequence length necessary for 100% accuracy). Simulations were allowed to continue for 5−10 images beyond this 100% accuracy point to ensure that no new false-positives would enter the results.

Faults types in the continuous models, however, could not be diagnosed with perfect accuracy because complete continuous ranges of sensitivities were detected as a single nominal fault type. Therefore, simulations of these defect models were run for a fixed number of images (typically 100 images) and the detection error was used as the metric of accuracy. At each pixel location, the detection error was calculated from the absolute difference between the injected fault-type sensitivity (e.g.. *F(0.578, 0)*) and the detected fault-type sensitivity (e.g. *F(0.55, 0)*). The mean and maximum values of detection error across the entire simulated sensor array were taken as the average and worst-case accuracy, respectively.

5.4.3 Source Images

Test images were selected randomly without replacement from a bank of 398 digital photographs, including a wide range of landscape, interior and people-based images taken by a typical amateur photographer (samples are shown in Figure 5.10).

Images were stored in uncompressed 8-bit greyscale format obtained by performing desaturation and bicubic downsampling operations on the high-resolution colour originals. Several resolutions of each picture were derived from the same source, enabling simulations on photos from approximately 1 M to 6 M pixel (MP) sizes in 1 MP increments. Only one resolution of pictures was used for a given simulation run, and most simulations were performed using the 1 MP set to reduce computation time. Colour imagers were simulated by splitting the original source images into red, green and blue colour planes to obtain three independent sets of monochrome images. Sensor orientation was maintained constant in the images so image rotation, which is performed automatically in many DSCs, was ignored.

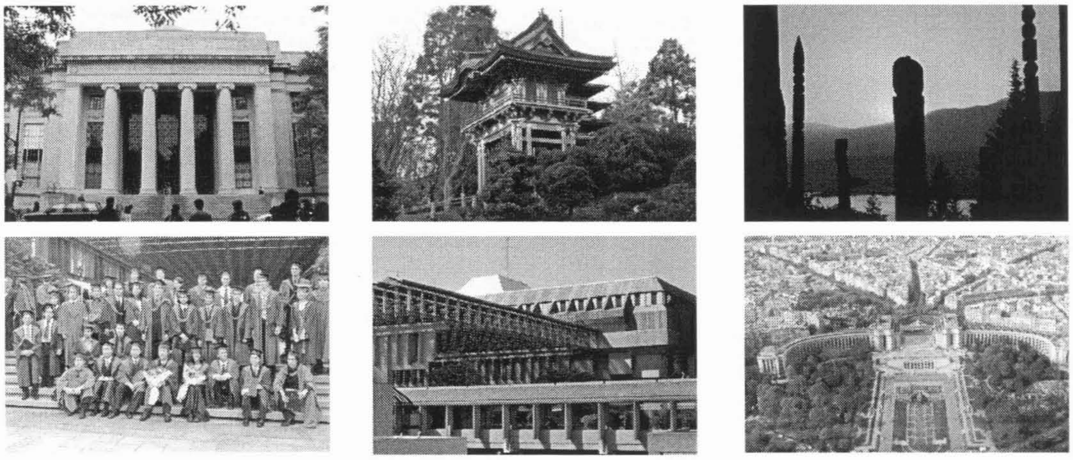


Figure 5.10. Sample images from simulation image bank.

5.4.4 Sensor Noise

Most simulations were performed using unmodified 1 MP images, which are smoothly varying and nearly noise-free due to the downsampling used to derive them. To simulate more realistic imaging systems, temporal noise was introduced into the test data in one of two ways. In the simplest method, greyscale versions of full resolution, 6 MP, images were passed to the algorithm, allowing the intrinsic sensor noise to affect the images. Noise power was varied by restricting simulations to include only images taken at particular camera gain settings (i.e. ISO Sensitivity). Because this gain is applied prior to any digital processing, the noise component from each pixel is also amplified, and increased ISO Sensitivity directly translates into increased noise power.

In a second set of simulations, a random value was added to every pixel output (not only faulty or good pixels) prior to processing by the detection algorithm. A zero-mean Gaussian random variable was used for simplicity, where the distribution variance was taken as the noise power.

5.5 Simulation Results

Results from simulations are now given to demonstrate how faults can be effectively identified in sensors affected by fault models of increasing complexity. In each case, the feasibility of fault identification is shown in the most basic experimental configuration that represents a simplified ideal system and then additional experiments are performed to show how the algorithm scales with various system parameters, such as sensor resolution and fault density. The goal of these tests is to confirm that such an algorithm could be successfully deployed in a real camera system.

5.5.1 Simple Stuck model

In the first set of simulations, the convergence rates of the image statistics and interpolation statistics methods were compared when implementing the *Simple Stuck* defect model. Table 5.8 gives a summary of some of the key simulation parameters.

Table 5.8. Simulation parameters used with the *Simple Stuck* fault model.

Parameter	*Value*
Number of iterations	100
Image resolution	1 MP
Fault density	0.002

Statistics of the convergence rate (the number of images required to achieve 100% accuracy), accumulated over all 100 simulations, are given in Table 5.9 for both algorithms.

Exact values were not available for the interpolation statistics method, but both algorithms were observed to perform very similarly with this simple defect model. In contrast to the results of [63, 64], this algorithm always correctly identified 100% of faults and correctly labelled 100% of all good pixels as well. Thus, the performance values presented below are a measure of the algorithm execution time rather than accuracy, which is already perfect at this model complexity.

Table 5.9.Algorithm performance using the Simple Stuck fault model.

	Convergence time (# of images analyzed)	
Algorithm	*Mean*	*Standard deviation*
Image statistics	3.45	3.91
Interpolation statistics (4NN averaging)	≈ 3.4	≈ 4

The histogram of convergence times in Figure 5.11 is representative of results obtained from both algorithms and shows that both identified all faults in less than 5 images 88% of the time. However, the outliers in the same histogram show that particularly poor fault placement, such as a stuck low fault affecting the dark regions of the first several images, can slow down identification and extend the required detection time to 25 pictures. Nonetheless, this rapid convergence speed clearly demonstrates that both the image statistics and interpolation statistics algorithms are well suited to identifying fully stuck faults because almost any application can readily produce 10 photographs. Note, however, that the image statistics implementation requires lower computational effort because it does not rely on any filtering steps beyond those already used by in-camera processing, making it ideal for low-cost applications.

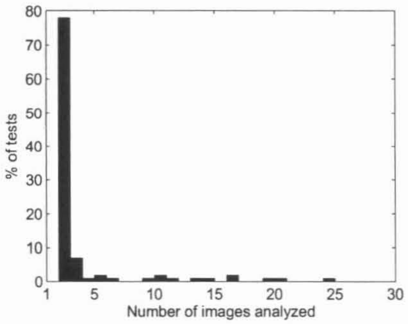


Figure 5.11. Histogram of convergence rates from 100 simulations of the *Simple Stuck* model, representing results of both Image Statistics and Interpolation Statistics algorithms.

5.5.2 FTAPS Stuck Model

Simple stuck fault types proved straightforward to diagnose because their outputs are fixed at one value in all images. However, complex abnormal sensitivity fault types in the *FTAPS Stuck* model are more challenging because their output value varies from image to image such that no fixed reference for the expected value is available. A second set of experiments were conducted to compare the performance of the image statistics and interpolation statistics identification algorithms in finding these faults. To better compare the effects of fault model complexity on performance, simulation parameters were maintained at the equivalent values to those specified in Table 5.10 (Note: the total fault density was increased in order to maintain the same density of individual fault types).

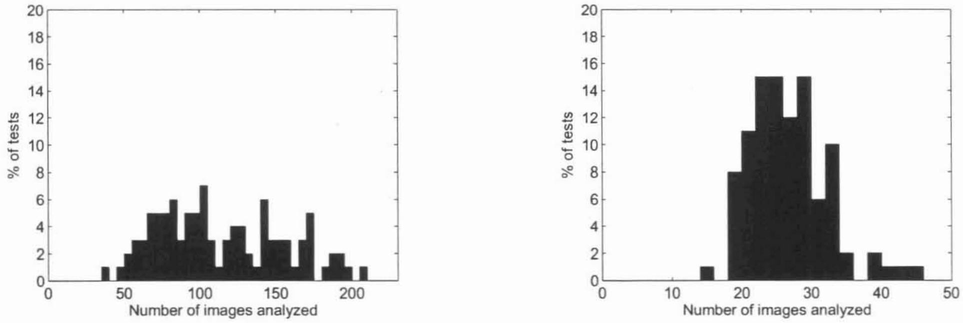
Table 5.10. Simulation parameters used with the *FTAPS Stuck* fault model.

Parameter	*Value*
Number of iterations	100
Image resolution	1 MP
Fault density	0.005
Interpolation type	4NN averaging
Darkfield used?	Yes

Simulated convergence rates from the first simulations of both the image statistics and interpolation statistics methods with *FTAPS Stuck* faults are summarized in Figure 5.12 and Table 5.11. The greatly increased convergence time values show that both algorithms require more information to pinpoint these complex light-sensitive faults compared to the *Simple Stuck* types. Here, the image statistics method analyzed 114 images on average, compared to only 3.5 with the *Stuck Simple* model. In the worst case, the image statistics algorithm needed data from more than 200 images before generating a perfect diagnosis, which is a prohibitively large data set for performing on-line identification.

Table 5.11. Algorithm performance using the *FTAPS Stuck* fault model.

	Convergence time (# of images analyzed)	
Algorithm	*Mean*	*Standard deviation*
Image statistics	114.3	41.0
Interpolation statistics (4NN averaging)	27.2	5.56



(a) Image statistics algorithm.

(b) Interpolation stats. algorithm (4NN averaging).

Figure 5.12. Histogram of convergence times from simulations of the FTAPS Stuck fault model using the (a) image statistics and (b) statistics algorithms.

Results from the interpolation statistics algorithm were more reasonable. On average, the algorithm analyzed 27.2 images with *FTAPS Stuck* faults, compared to 3.5 images in simulations with *Simple Stuck* faults. In the worst case, the simplest 4NN averaging interpolation statistics method required data from only 50 images to deliver a complete identification of all faults in the sensor, making this technique a viable option for implementing an in-camera fault identification system.

The disparity in convergence speed between the image statistics and interpolation statistics methods can best be understood be examining more closely the decision-making behaviour. In both cases, we can treat that decision as a continuous process of elimination that discards fault types when the observed pixel value is improbable. For example, the image statistics method will quickly determine that a pixel cannot be stuck high when a value of zero is observed. However, this image statistics process is slow to reach conclusions about complex abnormal-sensitivity faults because photographs tend to have a very wide distribution of pixel values. Thus, most fault types remain nearly equiprobable until a particular illumination value makes that combination of sensitivity and offset impossible. Then, the likelihood of that fault type becomes zero and it is effectively eliminated from the list possible candidates at the pixel location. In this way, incorrect fault types are “eliminated” as each pixel is exposed to a range of illumination values, but this process can require a large number of input photographs.

Conversely, the estimation step provides detailed information to the interpolation statistics method, resulting in a very narrow decision-making histogram (i.e. the $P_E(e)$ distribution). Once the estimated pixel value is known, only a small range of sensitivity or offset parameters can accurately map the observed value onto the expected value, eliminating a large number of other fault types from future consideration. As more images are tested, the remaining range of plausible fault types is further reduced, according to the accuracy of the interpolation method. Thus, the interpolation method statistics quickly converges on the correct fault type, justifying the added computation burden of performing the interpolation on each image.

5.5.2.1 Comparison of Interpolation Methods

Thus far, the 4NN averaging interpolation scheme has been used as a benchmark because it is the simplest of the schemes presented in Section 5.3.2.1. However, simulations were also run with the 8NN averaging scheme and the convergence time statistics from each run are presented in Table 5.12.

Table 5.12. Algorithm performance with varied interpolation methods.

	Convergence time	
	(# of images analyzed)	
Interpolation method	*Mean*	*Std. dev.*
4NN averaging	27.15	5.56
8NN averaging	17.58	7.87

The 8NN averaging scheme converges in less than 18 images on average and improves significantly on the simpler 4NN scheme, which takes, on average, 27 images to complete. The performance improvement is likely due to the smaller weighting given to each pixel, which in turn gives faulty pixels a smaller influence on interpolation results and allows quicker diagnosis of the good pixels around each fault. Here, the slightly reduced accuracy of 8NN averaging compared to 4NN (see Figure 5.5 on page 111 for a comparison) is insignificant because the interpolation error is much less than the error from faulty pixels themselves.

5.5.2.2 Application to Real-World Cameras

Once the interpolation statistics identification algorithm was demonstrated as a viable solution for locating defects in the *FTAPS Stuck* model, simulations were performed to evaluate how the software might perform in more realistic scenarios. The fault density was varied to represent a wide class of image sensors, and several resolutions of sensors were tested (i.e. those with increasing numbers of pixels and, consequently, image detail). Simulations were conducted with and without darkfield tests to determine whether this algorithm is equally applicable to cameras without a shutter, which cannot capture darkfields. The extension to colour imagers has not yet been made with the *FTAPS Simple* model, however. Both 4NN and 8NN averaging interpolation schemes were tested to find the range of situations in which both may be applicable.

Simulated algorithm performance is given in Table 5.13 as the fault density is increased from 5×10^{-5} to 0.05. We have already seen that these fault densities are orders of magnitude greater than those observed for in-field defects in commercial cameras, but the values may be applicable to some specialty imaging systems, such as infrared cameras. Nonetheless, even at these atypically high fault densities, both interpolation algorithms converged in less than 48 images in the worst-case. Typical convergence times were 27 and 18 images for the 4NN and 8NN schemes, respectively.

Table 5.13 shows that performance degraded as the defect density was increased, which can be attributed to two simple facts. First, because the distribution of $P_E(e)$ is derived at run-time from an image with faults, this function becomes less accurate as the number of faults increases. Furthermore, faults at certain locations will be more challenging to identify because they frequently affect regions of images with edges and will consequently give less accurate interpolation results. As fault density is increased, the probability of faults occupying these locations is increased, and thus the identification process is slowed down.

Table 5.13. Algorithm performance with increasing fault density.

	Convergence time (# of images analyzed)					
	4NN averaging			*8NN averaging*		
Fault density	*Mean*	*Std. dev.*	*Fail*	*Mean*	*Std. dev.*	*Fail*
5×10^{-5}	8.78	4.84	0	7.33	2.28	0
5×10^{-4}	17.09	6.58	0	10.61	6.82	0
5×10^{-3}	27.15	5.56	0	17.58	7.87	0
5×10^{-2}	-	-	100	86.42	58.13	74

Convergence sensitivity to image sensors with varied spatial resolving capability was simulated by processing images with the corresponding pixel count, as described in Section 5.4.3. Recall that each image was available in several resolutions such that all of the following simulations processed the same scene content but with varying levels of details. Because the interpolation step is so crucial to the algorithm, the level of image detail is expected to influence performance greatly. Table 5.14 lists the simulated convergence times for 1-3 MP sensors at two fault densities. Higher resolution sensors were not yet simulated because of the prohibitively long simulation time required to run these non-optimized implementations of the software. No failed tests were observed.

In all cases, the algorithm convergence time increased as sensor resolution was increased, but the resulting performance remained acceptable for most applications. In the worst-case scenario, the mean convergence time for 3 MP sensors using the 4NN scheme rose to 21.2 images from 17.1 images in the 1 MP case. Given the small magnitude of this performance decrease, it can likely be explained by the fact that higher resolution sensors will have more faults at a given fault density. Therefore, faults are more likely to occur in trouble locations, as discussed above, and the algorithm is slowed down.

Table 5.14. Algorithm performance with varied sensor resolutions.

		Convergence time (# of images analyzed)			
		4NN averaging		*8NN averaging*	
Sensor resolution	***Fault density***	*Mean*	*Std. dev.*	*Mean*	*Std. dev.*
1MP	5×10^{-5}	8.78	4.84	7.33	2.28
2MP		13.20	6.75	8.31	4.04
3MP		13.26	5.95	10.71	6.08
1MP	5×10^{-4}	17.09	6.58	10.61	6.82
2MP		20.23	7.05	14.27	8.45
3MP		21.20	6.47	15.51	8.58

Simulations were also performed with and without a darkfield test at the beginning of the image sequence. Such data is very simple to acquire in cameras equipped with a shutter and is expected to improve detection. However, simulations surprisingly showed that including darkfield data has negligible benefits in the identification of *FTAPS Stuck* faults (see Table 5.15). In fact, the darkframe contains little information because many of the faults in this model behave similarly under zero illumination. For example, both partial sensitivity faults and stuck low faults produce a black output without any light, and only the partially stuck high faults can be diagnosed this early in the process. Consequently, cameras without a mechanical shutter can employ this identification system without suffering a significant performance penalty.

Table 5.15. Algorithm performance with and without a darkfield image.

	Convergence time (# of images analyzed)			
	4NN averaging		*8NN averaging*	
Darkfield used?	*Mean*	*Std. dev.*	*Mean*	*Std. dev.*
Yes	8.78	4.84	7.33	2.28
No	13.20	6.75	8.31	4.04

All of these simulations thus far have been conducted without any clustering of faults. In a final simulation, this limitation was relaxed to allow multiple faults to strike the same 3×3 pixel space. However, the majority of these tests failed to converge on a correct defect map, illustrating a shortcoming of the interpolation statistics algorithm: the fault likelihood depends strongly on the pixel value estimate, $z_{i,j}$, which in turn depends on the validity of the neighbouring pixel outputs. When multiple pixels in the neighbourhood become defective, the interpolation error becomes very large and a correct diagnosis cannot be made. Consequently, the interpolation statistics algorithm should be replaced with the image statistics method when search for *FTAPS Stuck* faults in environments where clustering is known to be prevalent. Because image statistics does not estimate pixel values or compare neighbouring pixels in any way, this simpler algorithm is unaffected by clustering and can therefore provide a more accurate result, albeit after processing many more photographs.

The impact of estimation accuracy was observed to affect simulations of models that included additional partial sensitivity faults, such as the *F(0.25, 0)* and *F(0.75,0)* types. In these cases, the estimate $z_{i,j}$ at a location neighbouring a fault will clearly be incorrect. If only half-sensitivity faults are considered, that estimation error is not large enough to mistake the good pixel as *F(0.5,0)*, for example. However, as a finer granularity of sensitivity values is considered, the interpolation error can cause two closely spaced faults to be mistaken for each other, leading to false positive tests.

Despite this limitation, the interpolation statistics identification algorithm can clearly be combined with the Fault Tolerant Active Pixel Sensor design to create a complete self-healing camera system that can resist a wide range of faults in the field. Although hot pixels have not yet been considered in this statistical algorithm, ongoing research has already demonstrated that hot pixel faults can also be incorporated into the *FTAPS* model to make the camera truly resilient.

The following section demonstrates how the round-robin algorithm can overcome the limitations introduced by estimation error in the interpolation statistics algorithm.

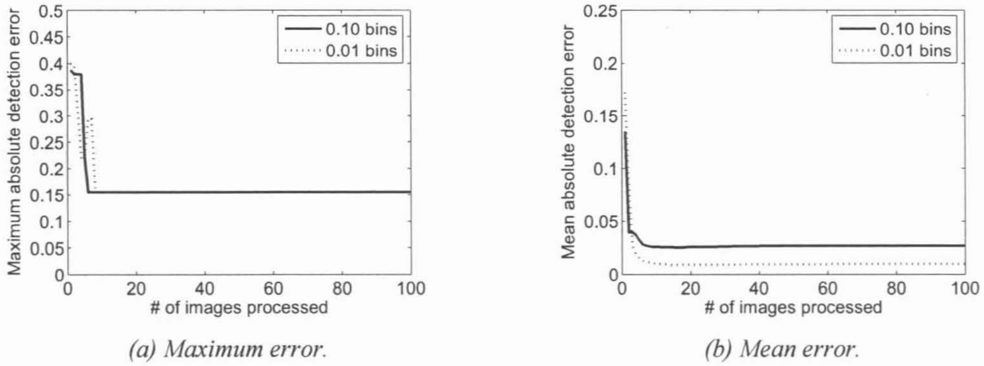
5.5.3 Continuous Fault Models

The interpolation statistics and round robin algorithms are now applied to the *Continuous* fault models that include faults with no offset and a continuous range of sensitivities. In effect, these following experiments simulate an in-field camera recalibration procedure. As discussed earlier, perfect identification of such faults is no longer feasible because of this continuous range of values. Therefore, detection error (absolute difference between injected and detected sensitivity) is now used as the performance metric. All simulations were run for 100 images, after which statistics of the error were accumulated and plotted before repeating the simulation for a total of 25 iterations. Once again, 1 MP source images were used. Table 5.16 lists the pertinent simulation parameters used in the remainder of the experiments.

Table 5.16. Simulation parameters used with the *Continuous* ranged fault models.

Parameter	***Value***
Number of iterations	25
Number of images tested	100
Image resolution	1 MP
Fault density	0.005
Darkfield used?	No
Cluster faults allowed?	No

We begin by evaluating the performance of the 4NN averaging interpolation statistics algorithm in identifying *Continuous Half* faults. Figure 5.13 shows the evolution of maximum and mean detection error throughout individual simulation runs. Results are plotted from two simulations: one with coarsely spaced 0.10 width bins and another simulation with more finely spaced 0.01 bins. From Figure 5.13(b) we see that mean detection error falls very quickly (in less than 10 images) and remains at this steady-state value until the simulation completes after 100 images. When using a model with 0.10 width bins, the algorithm settles to a mean error of ±0.026, while using finely spaced 0.01 bins results in a mean error of ±0.009. However, the large maximum error, which settles to ±0.16 in Figure 5.13(a), confirms the shortcomings of interpolation statistics. The aim of this algorithm is to improve image quality, and therefore occasional large errors cannot be tolerated as a tradeoff for correcting some smaller faults.



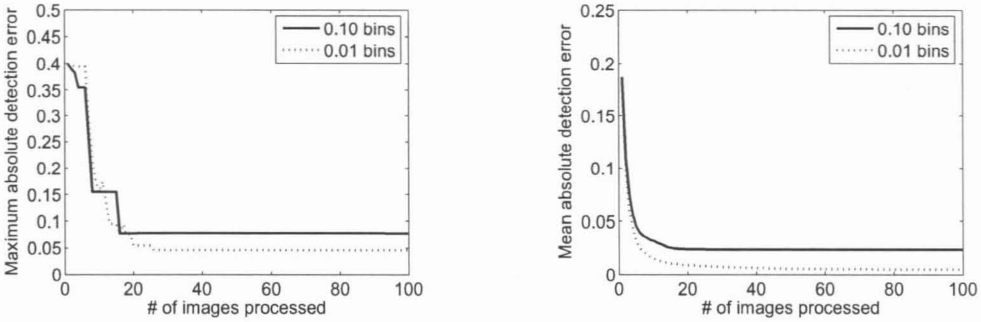
(a) Maximum error.

(b) Mean error.

Figure 5.13. Tracking detection error from the 4NN interpolation statistics algorithm using the *Continuous Half* defect model with 0.10 and 0.01 bin sizes.

Closer inspection of the algorithm workings suggested that occasional large errors in the estimation step cause a pixel to be incorrectly diagnosed. Furthermore, as very low sensitivity faults are introduced, the magnitude of these errors becomes very large near these faults, which in turn causes good pixels to be falsely identified as faulty and creates a large detection error. Therefore, although the interpolation statistics method is well suited to efficiently identifying simple, coarsely spaced fault types, another technique is required to mitigate occasional large estimation errors.

The majority voting nature of the round robin algorithm is expected to overcome the limitations discussed above because each estimation step is partitioned into four separate tests. Figure 5.14 shows the simulation results that demonstrate the success of round robin testing.



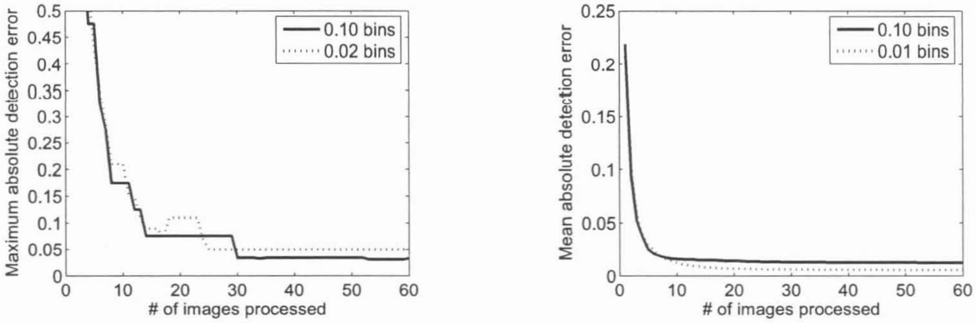
(a) Maximum error.

(b) Mean error.

Figure 5.14. Tracking detection error from the 4RR algorithm using the *Continuous Half* defect model with 0.10 and 0.01 bin sizes.

From Figure 5.14(b), we see that mean error once again settles to very small values: ±0.024 and ±0.005 for bin sizes of 0.10 and 0.01, respectively. However, the greatest improvement is seen in the maximum (worst-case) error plotted in Figure 5.14(a). With the round-robin algorithm, maximum error settles to about ±0.077 and ±0.046 for bin sizes of 0.10 and 0.01, respectively. Furthermore, this accuracy is achieved very rapidly. Maximum detection error reaches a steady state value within 40 images and mean detection error settles within 20 images, demonstrating that a very high accuracy can be achieved with a reasonable amount of test data.

As the model complexity is increased to include the *Continuous Full* faults, the round-robin algorithm continues to perform with excellent accuracy. Figure 5.15 shows that maximum and mean detection error both settle very quickly to the expected values. With 0.10 bins, maximum error settles to ±0.064 within about 40 images and mean error settles to 0.025 in less than 20 images. This closely corresponds to the ideal best-case scenario where accuracy is limited purely by bin granularity. That is, the worst-case error comes entirely from sensitivities at the bin edge being detected as the nominal value (e.g. both 0.901 and 0.999 faults are detected as the nominal 0.950 sensitivity, giving rise to ±0.05 error).



(a) Maximum error.

(b) Mean error.

Figure 5.15. Tracking detection error from the 4RR algorithm using the *Continuous Full* defect model with 0.10 and 0.02 bin sizes.

To better quantify the effects of defect model bin granularity, the *Continuous Full* model was simulated with 0.10, 0.05 and 0.02 bin sizes. Because of the large range of sensitivities included in this model, bin sizes smaller than 0.02 have prohibitively high computational requirements and were thus not tested. Simulation results, summarized in Table 5.17, show the expected linear relationship between bin size and mean error, but maximum error is only marginally affected. This limited improvement may once again be attributed to large errors in the estimation step that invariably occur in each image. Therefore, the limiting factor in maximizing accuracy is the estimation method and not necessarily the fault model granularity. Nonetheless, it should be noted that the round-robin algorithm accurately identifies abnormal sensitivity faults within ±0.05 even with the relatively coarse *Continuous Full-0.10* fault model that requires modest computation time. Once again, this accuracy corresponds closely to the best-case error achievable with this finite bin granularity. Given these encouraging results, all further simulations will focus on the *Continuous Full-0.10* model.

Table 5.17. Detection error as fault model bin size is varied. Values after 60 test images using 4RR algorithm with *Continuous Full* fault model.

	Detection error	
Model bin size	*Maximum*	*Mean*
0.10	0.064	0.024
0.05	0.034	0.013
0.02	0.050	0.006

5.5.3.1 Application to Real-World Sensors

Once again, several simulations were performed to verify the applicability of the round robin algorithm to a wide range of camera systems. Fault density and sensor resolution were varied to examine algorithm behaviour in a range of environments. In addition, simulations were run with varying noise powers to investigate the effects on estimation. Finally, colour image sensors were simulated by partitioning tests to operate individually on the red, green and blue colour planes.

In the first experiment, fault density was increased from 5×10^{-4} to 5×10^{-2} with the *Continuous Full-0.10* model (i.e. *Continuous Full* fault model with 0.10 bin sizes). The plots of that the detection error in Figure 5.16 show that the round-robin algorithm achieves an excellent and nearly constant accuracy at all tested fault densities. Both maximum and mean errors remain approximately constant near 0.06 and 0.04, respectively. This negligible impact of fault density on detection accuracy is explained again by the majority voting nature of the round robin algorithm which prevents good pixels from being labelled as faulty even when a faulty pixel is included in the estimation process. Note that this experiment was performed with artificially prohibited fault clusters. Nonetheless, at the moderate fault densities expected in commercial imagers, the likelihood of clustering is small.

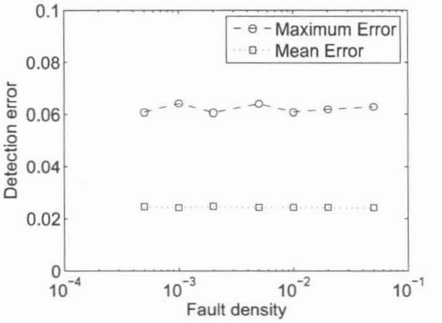


Figure 5.16. Detection error with increasing fault density using 4RR detection with *Continuous Full-0.10* fault model.

In addition, sensor resolutions ranging from 1 MP to 6 MP were tested with the same fault model and 1×10^{-3} fault density. The results are summarized in Figure 5.17, which shows that both maximum and mean detection accuracy remain approximately constant at all resolutions. Clearly, this algorithm is less sensitive to variations in image detail, which come from downsampling of photographs, because of the simple estimation method used.

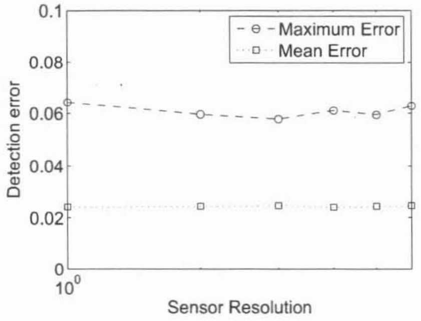
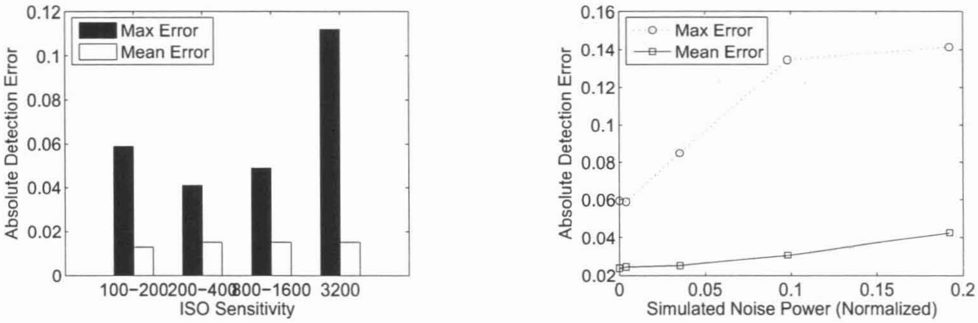


Figure 5.17. Detection error with increasing sensor resolution using 4RR detection with *Continuous Full-0.10* fault model.

5.5.3.2 Noisy Sensors

Noisy sensors were also simulated using both of the methods discussed in Section 5.4.4. Figure 5.18(a) shows the detection error achieved at four distinct noise levels. As expected, moderate noise powers, up to ISO 1600 on this particular camera (corresponding to a normalized luminance noise power of about 0.012), had little effect on detection error, and variations between simulations can be more likely attributed to the particular scene content. However, detection error in the noisiest images, at ISO 3200 (normalized luminance noise power of about 0.0164), was nearly double that of the moderate tests. Mean error was largely unaffected at any noise intensity, suggesting once again that increased temporal noise caused occasional estimation errors. At all moderate noise levels, however, detection accuracy remained at less than ±0.06, demonstrating that this algorithm can be applied to most mainstream sensors. Applications unable to tolerate the ±0.114 error introduced by the extremely noisy images could avoid this limitation by simply not processing them.



(a) Noisy source images.

(b) Simulated noisy sensor.

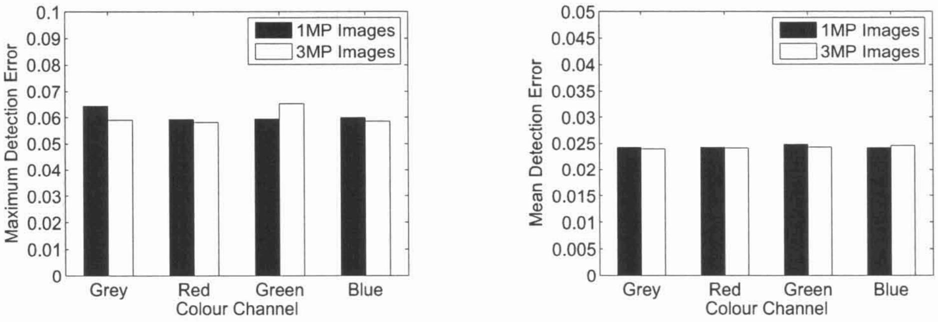
Figure 5.18. Detection error under varied sensor noise levels using 4RR algorithm with *Continuous Full-0.10* fault model. Noise was injected by using (a) noisy source images and (b) simulated noise injection.

For completeness, simulations were also performed using artificially-injected temporal noise in 1 MP images, which allows much larger noise powers to be considered. Figure 5.18(b) shows that the resulting detection accuracy degrades with increasing noise power, but the decline in performance is quite slow. Even at an unrealistic noise power of 0.20 (in normalized units), the 4RR algorithm achieves ±0.14 accuracy, providing further evidence that the identification method can be safely applied to most commercial sensors.

5.5.3.3 Colour Sensors

To demonstrate how this identification algorithm can be extended to colour imagers, simulations were performed on data extracted from the individual colour planes (red, green, and blue) of full-colour source images. Plots of the resulting detection accuracy are shown in Figure 5.19 along with results from greyscale tests presented thus far. Simulations were performed on both 1 MP and 3 MP images for completeness. From the plot of maximum error in the Figure below, we can see that the worst-case detection error remains approximately constant regardless of the colour channel being processed. In addition, the accuracy from colour images is about the same as that obtained from greyscale images in all of the simulations thus far.

These results help to validate this proposed approach to processing data from colour imagers. Although not implemented in time for this thesis, future versions of the software can expect to achieve excellent detection accuracy simply by separating test images into their constituent colour planes.



(a) Maximum detection error. *(b) Mean detection error.*

Figure 5.19. (a) Maximum and (b) mean detection error in a simulated full-colour camera using 4RR algorithm with *Continuous Full-0.10* fault model. Individual colour planes at 1MP and 3MP resolutions are tested.

Finally, the impact of clustering on detection was investigated in the following experiment. Using the 4 round-robin estimation scheme, up to two defects were placed at adjacent sites, while the 8RR scheme was examined with 2 and 4 adjacent defects in a 3×3 pixel block. Table 5.18 gives a summary of the maximum and mean detection error from simulations of both methods.

The results consistently show that both methods fail to provide reasonable detection accuracy when any level of clustering is allowed. Clearly, even the majority voting round robin method can be fooled when multiple faults impact the estimated value at a pixel site. The fact that mean error remains constant with defect cluster size indicates that these estimation errors only occur occasionally – most likely near fine details and edges in the image. However, such mistakes are unavoidable because a wide range of image types must be processed to apply this algorithm in the field. Future work that aims to correct errors in high fault density sensors should focus on overcoming these cluster errors using alternative approaches, such as an iterative algorithm.

Table 5.18. Simulated detection error from 4RR and 8RR algorithms with fault clustering.

	Detection error			
	4RR algorithm		***8RR algorithm***	
Fault cluster size (# of adjacent faults)	*Maximum*	*Mean*	*Maximum*	*Mean*
1	0.064	0.024	0.060	0.024
2	0.51	0.024	0.38	0.024
4	-	-	0.39	0.024

Despite these limitations under heavy fault clustering, the simulations reported in this section have served to validate this Bayesian inference-based algorithm as an effective tool for identifying a diverse set of fault types in widely varying conditions. Typically, sensors are affected by only moderate fault quantities, which the round robin method can fully calibrate with excellent accuracy (down to ±0.05 error in sensitivity). Even when clusters are considered, large detection errors are very rare and the vast majority of the fault map achieves the same accuracy. Furthermore, when the fault density becomes sufficiently high to produce large numbers of clusters, much of the information originally captured by the sensor will be discarded by the correction algorithm, in which case hardware fault tolerance or outright replacement should be considered as alternatives to this software-only approach.

5.6 Summary

Three closely-related algorithms have been described for locating faults and identifying their type, all of which use Bayesian inferences to match observed pixel response to mathematical fault behaviour models. The simplest technique, called the image statistics method, looks at the histogram from each image to find when resulting pixel values are unlikely. In the interpolation statistics method, the expected pixel value is estimated using interpolation and is compared to the observed pixel value to find the most likely fault model. The round robin methods partition interpolation into separate steps to effectively form a majority-voting scheme with the aim of avoiding catastrophic errors in the estimation step.

Monte Carlo simulations were performed on all three algorithms to determine their accuracy and speed in identifying particular fault types. The image statistics method was able to identify fully stuck faults in less than 20 images with 100% accuracy, while the more complex partial sensitivity faults in the FTAPS model required up to 220 images to be fully diagnosed. The interpolation statistics method improved on this performance by identifying FTAPS faults in less than 60 images every time. Extended simulations showed that the interpolation-based identification scheme could be applied to a wide range of sensors without a significant performance penalty. Sensors up to 3 MP resolution were tested with fault densities extending to an unrealistic 5×10^{-3}.

Although the interpolation statistics method was not effective at identifying continuously distributed faults due to large errors in the estimation step, round robin partitioning enabled a very accurate calibration of abnormal sensitivity faults. In the best case, pixel sensitivities were identified within an error of ±0.034, and average results with only moderate algorithm complexity (i.e. 4RR method with 0.10 bin size fault model) yielded a maximum error of no more than ±0.064. Further simulations demonstrated that the round-robin methods could be extended to sensors up to at least 6 MP resolution at any fault density without degrading performance. Furthermore, the algorithm performed equally well on three-colour, red-green-blue sensors. Clustering of multiple faults in a single 3×3 continues to limit accuracy in some cases, but such clustering would not be significant in sensors with any reasonable fault density arising either in the field or during manufacturing.

Thus, a novel and robust fault identification scheme has been presented and shown to be effective at finding the locations and types of faults distributed throughout electronic image sensors. This software may be readily deployed using existing sensor hardware to improve the robustness of cameras in the field without adding significant cost to the camera. In some applications, this system may even be used to replace some stages of manufacture-time calibration to further reduce production costs or improve yield.

CHAPTER 6

CONCLUSIONS

Solid-state image sensors have reached ubiquitous status, promising to transform aspects of daily life ranging from traditional photography to automobile safety. This widespread reliance on image sensors necessitates a better understanding of their long-term reliability and systems to mitigate potential failures. The aim of this thesis was to address that need in three ways: by characterizing in-field defects; demonstrating how the FTAPS can mitigate the most common fault type; and developing a novel fault identification and correction algorithm for all image sensors.

6.1 Fault Characterization

In Chapter 3, this thesis demonstrated concretely how a straightforward suite of darkfield, uniform lightfield, and photograph analysis procedures could identify and track in-field defect growth in commercial cameras. While specialized scientific cameras have been the subject of limited investigations in the past, this work is the first to make publicly available a detailed study of aging effects on ordinary solid-state imagers in readily-available cameras.

As a part of this research, in-field defect development was characterized in high-end digital SLR cameras. Contrary to commonly-cited anecdotal reports, quantitative tests found no fully stuck or abnormal sensitivity faults in any of the 11 cameras tested. However, hot pixels were determined to be the dominant fault type, with 101 hot pixels identified across 9 cameras. Of all faults, 8 were previously-unknown partially-stuck hot pixels, which degrade images at any exposure duration. Moreover, extended tests on a single camera found that 3 of 15 hot pixels exhibited random telegraph signal behaviour that would confuse most correction algorithms because of their frequently-changing dark current levels.

A detailed statistical analysis methodology was also developed to examine the spatial and temporal behaviour of in-field pixel faults beyond the simple fault-counting methods typically found in the literature. Maps of fault locations were generated and the inter-defect distances computed, which indicated that faults are distributed uniformly across sensors and do not form smaller clusters. All hot pixels developed as single-pixel faults, suggesting that the physical defects are small and point-like, confined to less than 0.15% of the total pixel area.

Temporal data was extracted from series of regular photographs, which showed that hot pixels develop suddenly and do not heal over time. In addition, the growth of new faults over time was found to be a continuous process, producing hot pixels in one test camera at a steady rate of about 5.5 hot pixels per year. The observed time intervals between the arrivals of subsequent faults suggests they are produced by a Poisson random process.

Together, these clues indicate that the in-field faults were the result of an external stress, rather than intrinsic material degradation. The observed behaviour is consistent with the findings of previous radiation experiments by researchers for space-borne cameras, suggesting that hot pixels are specifically caused by silicon bulk displacement damage arising from single radiation events. Although the test cameras had operated only in benign environments, they were constantly bombarded by terrestrial cosmic rays consisting primarily of high-energy protons and neutrons impinging at a sufficient rate to cause about 5.5 annual faults, on average.

Six or fewer faults developed annually may seem insignificant, but these blemishes mark every image captured by the camera and quickly stand out in ordinary photographs. Furthermore, because cosmic ray intensities vary with geographic location and altitude, cameras subjected to more harsh environments, like those frequently taken on trans-continental flights, can expect to develop orders of magnitude more hot pixels. Similarly, large area imagers, which will become more commonplace as semiconductor manufacturing processes improve, can expect to develop even more hot pixels with a very noticeable impact on image quality.

Overall, this study has quantified, for the first time in public literature, the significance of in-field faults in multiple common silicon solid-state imagers. The quantitative fault growth evidence presented here can aid future engineers in designing sensors to estimate and mitigate the impact of hot pixels.

Moreover, the simple testing and analysis procedures developed here can be readily extended to collect data from a much larger and diverse sample set. For example, future studies can re-use these test processes to calibrate many more DSLR cameras to improve the statistical significance of the results. Alternately, the same process can be very easily adapted to other classes of imagers, such as cell phone cameras or more compact consumer-grade cameras. In either case, this work has helped to lay the groundwork for a straightforward user-centric imager calibration process.

6.2 Hot Pixel Mitigation with Fault Tolerant Active Pixel Sensors

Hot pixel faults not only add false bright spots to images, they also degrade a pixel's dynamic range by causing pixels to saturate earlier than expected. Consequently, simple darkframe subtraction is insufficient for correcting these faults. Chapter 4 described how the redundant architecture of the FTAPS could overcome this limitation by isolating hot defects to a portion of the in-pixel circuitry. The FTAPS partitions the standard 3-transistor APS architecture into two parallel sub-pixels, achieving pixel-level fault tolerance with minimal overhead. When afflicted by a hot defect, only one half of the total pixel saturates due to the enhanced dark current, while the other sub-pixel continues to operate unaffected. Thus, the output current summed from each sub-pixel includes both light-dependant and offset components.

This thesis developed an accompanying software algorithm to recover the desired light-dependant portion of the FTAPS signal by estimating the amount of dark signal accumulated in the hot sub-pixel. Stored calibration curves of the non-linear pixel response are then referenced to solve for the correct output signal. The accuracy of the system was evaluated experimentally in two stages. Initially, hot pixels were optically emulated in a standard FTAPS array by preferentially illuminating one sub-pixel using a focused laser, inducing an additional "leakage current." At this first stage, charge blooming between sub-pixels limited the correction algorithm accuracy to ±0.05 normalized error in 96% of cases. In a second round of testing, hot pixels were emulated electrically to enable automated testing. A custom FTAPS array was designed with in-pixel control transistors to induce and control a leakage current in parallel with the photocurrent. In these experiments, the recovery algorithm computed the true scene illumination within ±0.05 normalized error in 98% of cases. Extreme photo and dark currents prevented better accuracy in the electrical tests, which could be improved in future designs by limiting intrinsic dark current.

Thus, a complete, self-healing, fault tolerant imaging solution has been developed and validated. While such a system may be unnecessary given the relatively small fault quantities observed in characterization experiments, the FTAPS will be valuable in safety-critical and specialized long-term applications. For example, automobile safety systems must reliably withstand a variety of harsh environment conditions for periods up to 20 years, which would lead to possibly hundreds of hot pixels and other faults that the FTAPS can effectively mitigate. Similarly, high-altitude imaging systems in aircraft will be subjected to cosmic ray intensities orders of magnitude greater than imagers on the ground, potentially creating many dozens of hot pixels per year. The FTAPS can effectively mitigate the effects of those large fault quantities while maintaining nearly the same image quality as a new, undamaged sensor.

6.3 In-Field Fault Identification Algorithm

Traditional methods to locate faults in imagers are complex, labour intensive and time consuming, and are thus not suitable to automatic in-field identification. However, for a hardware fault tolerance solution like the FTAPS to be a viable stand-alone, self-healing imaging solution, it must be able to locate and identify faults in the field without manual user interaction. To serve that need, this thesis described three related Bayesian algorithms for locating and quantitatively identifying fault types by processing only regular photographs from cameras. Algorithm performance was verified with Monte Carlo simulations of the imaging system.

The simplest image statistics algorithm identified fully stuck faults after analyzing only 5 images, making it ideal for correcting even manufacture-time faults in inexpensive mobile cameras. An interpolation-based algorithm, meanwhile, was successfully able to identify fully stuck, partially stuck, and half-sensitivity faults in less than 60 images. When combined with the fault tolerant active pixel sensor architecture, this software creates a robust imaging system that can mitigate faults online, in the field, without user interaction or knowledge. Furthermore, both algorithms achieved 100% accuracy, unlike competing methods. The third method, termed the round-robin algorithm, identified large quantities of abnormal sensitivity pixels within ±0.06 of their true sensitivity after processing only 40 images.

This novel software would be combined with existing sensor hardware to improve the robustness of cameras in the field without adding complexity or cost to the imaging systems. In cost-sensitive applications, portions of the factory calibration can now be skipped. Furthermore, these algorithms can be used to automate faulty-pixel calibration in the field, eliminating the need for extensive user interaction and greatly broadening the range of devices that can be tested.

In addition, this identification algorithm allows a much wider exploration of defects in the field. For example, the methods described here can already be used to determine the behaviour of anecdotal fully stuck faults by simply analyzing photograph sequences submitted by users in the field.

6.4 Future Work

Each of the three components of this work have been preliminary in nature and would benefit from further research. Work is already underway to extend the in-field fault identification algorithm to detect the most commonly found in-field fault type: hot pixels[66]. Detecting the presence and magnitude of hot pixels would ordinarily be difficult because of the dark current variations that occur from image to image due to exposure duration and temperature variations. However, the algorithm could be simplified to look only for pixels with an obvious offset, and the probability likelihood could be readily modified to simultaneously consider a complete range of offset values in a single fault type.

The study of in-field fault characteristics in Chapter 3 would benefit greatly from an increased sample size to improve the statistical relevance of the results. Testing many cameras would also allow the radiation effects theory to be better validated by incorporating geographical information from a more diverse dataset. However, the time-consuming and complex nature of the tests makes it difficult to find users willing to contribute data from their cameras. Thus, future research should focus on implementing an automated camera calibration test suite that can be run from users' computers at the click of the mouse. Publicly-available software libraries, such as the gPhotoLib suite[67] could be readily used to control a range of commercial cameras while custom software would prompt users to perform simple tasks.

Similarly, the on-line fault identification algorithm could readily be extended to perform fault characterization without any significant user interaction at all. By analyzing only photographs, the calibration step could be eliminated, making it more likely that a very large sample of camera data could be acquired. This work is already under way with software being developed for [66], which would automatically trace defect growth over time, transmitting only the resulting data instead of entire photographs, and thereby eliminating privacy and copyright issues with users images.

The FTAPS has been extensively tested and characterized by other students, while this work demonstrated its usefulness in mitigating hot pixels. To fully validate this recovery operation, however, devices should be subjected to neutron bombardment to induce and examine displacement-damaged hot pixels. Designing sensors for imaging-specific fabrication processes, such as those offered by Taiwan Semiconductor Manufacturing Corporation would also help to demonstrate the commercial value of this architecture.

REFERENCES

[1] Camera and Imaging Products Association, "Production, Shipment of Digital Still Camera, July, January-July 2007," CIPA, July, 2007. http://www.cipa.jp/english/data/pdf/d_200707.pdf

[2] Camera and Imaging Products Association, "2008-2010 Outlook on the Shipment Forecast by Product-Type Concerning Cameras and Related Goods," CIPA, Tokyo, Japan, Jan. 28, 2008. http://www.cipa.jp/english/pdf/press080129_e.pdf

[3] *Iteris Inc. Home Page,* Oct. 19, 2007; http://www.iteris.com/av/na_test.html.

[4] General Motors, *Buick Offers New GM Safety-Enhancing Technologies Designed To Help Drivers Stay In Their Lane, Avoid Lane-Change Collisions,* Oct. 19, 2007; http://www.gm.com/corporate/responsibility/safety/news/2007/tech_safety_041807_content.jsp.

[5] Infinity Today, *Advanced Lane Departure Warning System,* Oct. 19, 2007; http://www.infiniti.com/infiniti_today/latest_news/Advanced_Lane_Departure_Warning_System.html.

[6] Sensata Technologies, "Automotive Cameras for Safety and Convenience Applications," June, 2007.

[7] Braintech Inc., *eVisionFactory product description.* Nov. 2, 2007; http://www.bnti.com/evf.html.

[8] DPix Inc., "T60 Specification Sheet," Nov. 16, 2006. http://www.dipix.com/products/pdf/T60SpecSheet.pdf

[9] *Ellips Vision Technology,* Nov. 2, 2007; http://www.ellips.nl/home/ellips_vision_technology.html.

[10] Sensata Technologies, "Auto Brite Imaging Technology: Wide Dynamic Range for Automotive Machine Vision," June, 2007.

[11] H. Takahashi, M. Kinoshita, K. Morita, T. Shirai, T. Sato, T. Kimura, H. Yuzurihara, S. Inoue, S. Matsumoto and S. D. D. Center, "A 3.9 µm pixel pitch VGA format 10-b digital output CMOS image sensor with 1.5 transistor/pixel," *IEEE Journal of Solid-State Circuits*, vol. 39*(12)*, pp. 2417-2425, 2004.

[12] J. Bogaerts, B. Dierickx, G. Meynants and D. Uwaerts, "Total dose and displacement damage effects in a radiation-hardened CMOS APS," *IEEE Transactions on Electron Devices*, vol. 50*(1)*, pp. 84-90, 2003.

[13] Y. Chen, S. M. Guertin, M. Petkov, D. N. Nguyen and F. Novak, "A chip and pixel qualification methodology on imaging sensors," *Proceedings IEEE International Reliability Physics Symposium,* pp. 435-439, Apr. 2004.

[14] T. R. Oldham and F. B. McLean, "Total ionizing dose effects in MOS oxides and devices," *IEEE Transactions on Nuclear Science*, vol. 50*(3 Part 3)*, pp. 483-499, 2003.

[15] G. H. Chapman and Y. Audet, "Creating 35 mm camera active pixel sensors," *Proceedings IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems,* pp. 22-30, Albuquerque, NM, Oct. 1999.

[16] S. Djaja, "CMOS Active Pixel Sensors that Incorporate Concepts of Fault-Tolerance and Active Background Illumination Discrimination," M.A.Sc. thesis, Simon Fraser University, Burnaby, BC, 2006.

[17] D. Y. H. Cheung, "CMOS Active Pixel Sensor for Fault Tolerance and Background Illumination Subtraction," M.A.Sc. thesis, Simon Fraser University, Burnaby, BC, 2005.

[18] M. L. La Haye, G. H. Chapman, C. Jung, D. Y. H. Cheung, S. Djaja, B. Wang, G. Liaw and Y. Audet, "Characteristics of fault-tolerant photodiode and photogate active pixel sensor (APS)," *Proceedings IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems,* pp. 58-66, Cannes, France, Oct. 2004.

[19] M. L. La Haye, C. Jung, D. Chen, G. H. Chapman and J. Dudas, "Fault tolerant active pixel sensors in 0.18 and 0.35 micron technologies," *Proceedings IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems,* pp. 448-456, Washington, DC, Oct. 2006.

[20] C. G. Jung, "Characterization of Fault Tolerant and Duo-Output Active Pixel Sensors," M.A.Sc. thesis, Simon Fraser University, Burnaby, BC, 2007.

[21] M. L. La Haye, "Novel fault tolerant active pixel sensors," M.A.Sc. thesis, Simon Fraser University, Burnaby, BC, 2008.

[22] C. Jung, M. H. Izadi and M. L. La Haye, "Noise analysis of fault tolerant active pixel sensors," pp. 140-148, Monterey, CA,

[23] M. L. La Haye, C. Jung, M. H. Izadi, G. H. Chapman and K. S. Karim, "Noise analysis of fault tolerant active pixel sensors with and without defects," *Proceedings of the SPIE,* vol. 6068, San Jose, CA, Jan. 2006.

[24] J. Nakamura, "Basics of image sensors," in *Image Sensors and Signal Processing for Digital Still Cameras* J. Nakamura, Ed. Boca Raton, FL: Taylor & Francis Group, 2006, pp. 55-59.

[25] G. P. Weckler, "Operation of pn Junction Photodetectors in a Photon Flux Integrating Mode," *IEEE Journal of Solid-State Circuits*, vol. 2*(3)*, pp. 65-73, 1967.

[26] N. Teranishi, A. Kohono, Y. Ishihara, E. Oda and K. Arai, "No image lag photodiode structure in the interline CCD image sensor," *Proceedings International Electron Devices Meeting,* vol. 28, pp. 324-327, Dec. 1982.

[27] P. J. W. Noble, "Self-scanned silicon image detector arrays," *IEEE Transactions on Electron Devices*, vol. 15*(4)*, pp. 202-209, 1968.

[28] E. R. Fossum, "Active pixel sensors: Are CCDs dinosaurs?" *Proceedings of the SPIE,* vol. 1900, San Jose, CA, Jan. 1993.

[29] D. James, L. Klibanov, G. Tompkins and S. J. Dixon-Warren, "Inside CMOS Image Sensor Technology," Chipworks Inc., 2006.

[30] W. S. Boyle and G. E. Smith, "Charge-coupled semiconductor devices," *Bell Systems Technology Journal*, vol. 49, pp. 587-593, 1970.

[31] T. Yamada, "CCD image sensors," in *Image Sensors and Signal Processing for Digital Still Cameras* J. Nakamura, Ed. Boca Raton, FL: CRC Press, Taylor and Francis Group, 2006, pp. 95-141.

[32] I. Baharav, R. Kakarala, X. Zhang and D. W. Vook, "Bad pixel detection and correction in an image sensing device," U.S. Patent 6737625, May 18, 2004.

[33] S. Fumihiko Fumihiko and T. Asaida, "Image defect correcting circuit for a solid state imager," U.S. Patent 5144446, Sep. 1, 1992.

[34] D. J. Gusmano, "Pixel by pixel offset and gain correction in analog data from scanning arrays," U.S. Patent 5204761, Apr. 20, 1993.

[35] F. Sudo and T. Kihara, "Apparatus for correcting for defective pixels in an image sensor," U.S. Patent 5381175, Jan. 10, 1995.

[36] Y. Takemura, "Blemish compensating system for a solid state image pick-up device," U.S. Patent 4237488, Dec. 2, 1980.

[37] J. Yen, D. R. Tretter and T. Nguyen, "Correcting defect pixels in a digital image," U.S. Patent 6724945, Apr. 20, 2004.

[38] E. A. Amerasekera and F. N. Najm, *Failure Mechanisms in Semiconductor Devices,* 2nd ed., New York, NY: John Wiley & Sons, 1997.

[39] K. Chakraborty and M. Pinaki, *Fault-Tolerance and Reliability Techniques for High-Density Random-Access Memories,* Upper Saddle River, NJ: Prentice Hall, 2002.

[40] G. Di Giacomo, *Reliability of Electronic Packages and Semiconductor Devices,* McGraw-Hill, 1996.

[41] S. K. Tewksbury, *Wafer-Level Integrated Systems: Implementation Issues,* Norwell, MA: Kluwer, 1989.

[42] J. C. Pickel, A. H. Kalma, G. R. Hopkinson and C. J. Marshall, "Radiation effects on photonic imagers-a historical perspective," *IEEE Transactions on Nuclear Science*, vol. 50*(3)*, pp. 671-688, 2003.

[43] J. R. Srour, C. J. Marshall and P. W. Marshall, "Review of displacement damage effects in silicon devices," *IEEE Transactions on Nuclear Science*, vol. 50*(3 Part 3)*, pp. 653-670, 2003.

[44] J. Bogaerts, B. Dierickx, R. Mertens and H. IMEC, "Enhanced dark current generation in proton-irradiated CMOS activepixel sensors," *IEEE Transactions on Nuclear Science*, vol. 49*(3)*, pp. 1513-1521, 2002.

[45] M. Cohen and J. P. David, "Radiation-induced dark current in CMOS active pixel sensors," *IEEE Transactions on Nuclear Science*, vol. 47*(6)*, pp. 2485-2491, 2000.

[46] A. M. Chugg, R. Jones, M. J. Moutrie, J. R. Armstrong, D. B. S. King and N. Moreau, "Single particle dark current spikes induced in CCDs by high energy neutrons," *IEEE Transactions on Nuclear Science*, vol. 50*(6)*, pp. 2011-2017, 2003.

[47] H. L. Hughes and J. M. Benedetto, "Radiation effects and hardening of MOS technology: devices and circuits," *IEEE Transactions on Nuclear Science*, vol. 50*(3)*, pp. 500-521, 2003.

[48] G. R. Hopkinson, "Radiation effects in a CMOS active pixel sensor," *IEEE Transactions on Nuclear Science*, vol. 47*(6)*, pp. 2480-2484, 2000.

[49] H. H. K. Tang and K. P. Rodbell, "Single-event upsets in microelectronics- Fundamental physics and issues," *Material Research Society Bulletin*, vol. 28*(2)*, pp. 111-116, 2003.

[50] F. W. Sexton, "Destructive Single-Event Effects in Semiconductor Devices and ICs," *IEEE Transactions on Nuclear Science*, vol. 50, pp. 603-621, Jun., 2003.

[51] J. Bogaerts, B. Dierickx and R. Mertens, "Random telegraph signals in a radiation-hardened CMOS active pixel sensor," *IEEE Transactions on Nuclear Science*, vol. 49*(1)*, pp. 249-257, 2002.

[52] W. C. McColgin, C. Tivarus, C. C. Swanson and A. J. Filo, "Bright-pixel defects in irradiated CCD image sensors," *Proceedings Materials Research Society Symposium,* vol. 994,

[53] A. J. P. Theuwissen, "Influence of terrestrial cosmic rays on the reliability of CCD image sensors," *International Electron Devices Meeting Technical Digest,* pp. 811-814, Dec. 2005.

[54] J. F. Ziegler, "Terrestrial cosmic rays," *IBM Journal of Research and Development*, vol. 40*(1)*, pp. 19-39, 1996.

[55] J. F. Ziegler, "Terrestrial cosmic ray intensities," *IBM Journal of Research and Development*, vol. 42*(1)*, 1998.

[56] T. C. May and M. H. Woods, "Alpha-particle-induced soft errors in dynamic memories," *IEEE Transactions on Electron Devices*, vol. 26*(1)*, pp. 2-9, 1979.

[57] I. H. Hopkins and G. R. Hopkinson, "Further measurements of random telegraph signals in protonirradiated CCDs," *IEEE Transactions on Nuclear Science*, vol. 42*(6)*, pp. 2074-2081, 1995.

[58] I. Koren and Z. Koren, "Defect tolerance in VLSI circuits: techniques and yield analysis," *Proceedings of the IEEE*, vol. 86*(9)*, pp. 1819-1838, 1998.

[59] Semiconductor Insights Inc., "Device Quick Look: Canon CMOS Image Sensor Technology in EOS 10D Camera," 2003.

[60] I. Takayanagi, "CMOS image sensors," in *Image Sensors and Signal Processing for Digital Still Cameras* J. Nakamura, Ed. Boca Raton, FL: CRC Press, Taylor and Francis Group, 2006, pp. 143-178.

[61] H. I. Kwon, I. M. Kang, B. G. Park, J. D. Lee and S. S. Park, "The analysis of dark signals in the CMOS APS imagers from the characterization of test structures," *IEEE Transactions on Electron Devices*, vol. 51*(2)*, pp. 178-184, 2004.

[62] R. Rambaldi, M. Tartagni and A. H. Kramer, "Pixel Correction System and Method for CMOS Imagers," U.S. Patent 6618084, Sep. 9, 2003.

[63] B. P. Jin, N. George and K. M. M. C. Yeary, "Modeling and analysis of soft-test/repair for CCD-based digital X-ray systems," *Instrumentation and Measurement, IEEE Transactions on*, vol. 52*(6)*, pp. 1713-1721, 2003.

[64] Y. P. Tan and T. Acharya, "A robust sequential approach for the detection of defective pixels in an image sensor," *Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing,* vol. 4, pp. 2239-2242, Phoenix, AZ, Mar. 1999.

[65] I. Koren, G. Chapman and Z. Koren, "Advanced fault-tolerance techniques for a color digital camera-on-a-chip," *Proceedings IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems,* pp. 3-10, San Francisco, CA, Oct. 2001.

[66] J. Leung, J. Dudas, G. H. Chapman, Z. Koren and I. Koren, "Characterization of pixel defect development during imager lifetime," *Proceedings of the SPIE,* vol. 6816, San Jose, CA, Jan. 2008.

[67] *gPhoto 2 Digital Camera Software Home Page,* Jan. 9, 2008; http://www.gphoto.org/.

APPENDICES

Appendix A: User Camera Calibration Procedure

The following represents that instructions provided to camera users on how to perform simple darkfield and brightfield calibration procedures on their cameras. Also specified is the information requested about the origins and lifetime of the camera.

Darkfield Calibration

Darkfield images are manually captured by the photographer. In addition, we ask that you send us a small number of photos you recorded when you first purchased the camera. These extra photos will tell us whether the defective hot pixels were present when the camera was manufactured or they developed over time.

Camera Setup

The camera should be configured according to the settings in the following table:

Table 1. Suggested camera configuration for darkframe calibration.

Parameter	*Value*
Sensitivity	ISO 400
Output Mode	RAW images (not JPEG)
Capture Mode	Manual (or Exposure Priority / Tv)
Noise Reduction	Disable any "long-exposure noise reduction" settings
PC Connection	Either tethered or hand-held operation is okay
Auto Focus	Off (i.e. set to Manual Focus)
Aperture	Any

Darkframe Capture

Follow the procedure below to capture a sequence of darkframe calibration photos:

1) Place the lens-cap on the camera.
2) Cover the viewfinder with a heavy cloth or a bag, etc.
3) Set the exposure duration to 2 seconds.
4) Capture one photo.
5) Wait for the capture and save process to complete (i.e. wait for the "CF Busy" indicator to turn off).
6) Repeat steps 2-5 for each of the exposure durations listen in Table 2.

Desired exposure durations are given for both possible camera configurations for convenience.

Table 2. Suggested exposure durations for darkframe capture.

Exposure Duration	
Camera set to ½ stop increments	*Camera set to 1/3 stop increments*
2 s	2 s
1 s	1 s
0.800 s	0.800 s
0.500 s	0.500 s
0.300 s	0.300 s
1/4 s	1/5 s
1/5 s	1/8 s
1/8 s	1/13 s
1/11 s	1/20 s
1/15 s	1/30 s

Early Lifetime Photos

We need a few (3-4) photos taken near the time when you first got your camera. The pictures should have the following characteristics:

- Any format is ok (RAW, JPEG, TIFF)
- Full sensor size (not cropped or enlarged)
- Exposure duration should be EQUAL OR LONGER THAN 1/10s
- If you post-process the photos, please leave EXIF and other metadata in the file.
- Any image content is suitable.

These pictures will never be published, sold, displayed or otherwise shared with anyone. They will be stored on a secure server with no public access.

Flatfield Calibration

Flatfield images are again manually captured by the photographer. They will help calibrate the light sensitivity of the sensor and detect if any subtle changes happen over time. We do this by starting with a roughly uniform light source and adding make-shift filters / diffusers to cut down the intensity and to distribute the light more evenly.

Camera Setup

Once again, please configure the camera according to the settings below:

Table 3. Suggested camera configuration for flatfield calibration.

Parameter	***Value***
Sensitivity	ISO 400
Output Mode	RAW images (not JPEG)
Capture Mode	Manual
Lens	Remove (i.e. camera lightbox exposed)
Noise Reduction	Disable any "long-exposure noise reduction" settings
PC Connection	Either tethered or hand-held operation is okay
Auto Focus	Off (i.e. set to Manual Focus)
Exposure Duration	Any

Note: we remove the lens to distinguish between sensor defects and blemishes or dust that may be present on the lens. If possible, please try to ensure that the sensor surface is free of dust.

Flatfield Capture

The following procedure assumes you do not have access to more sophisticated lighting equipment that is typically used to create a uniform field of light (or don't have the time to use it). If you do can use such equipment, we would welcome the additional information.

To capture flatfield images, a uniformly illuminated surface must be photographed under several levels of illumination brightness. The challenge is to distinguish between sensor defects and those on the lens. Without specialized equipment, the simplest method to accomplish this is the following:

1) Find a location with reasonably uniform ambient lighting (i.e. overhead fluorescent lighting in an office setting works well, outside on a sunny day will work as well).
2) Cut 2 sheets of standard white letter paper into about 16 rectangular pieces that are just large enough to cover the lens-mount on your camera. These paper pieces will be stacked to form a makeshift variable diffuser.
3) Remove the lens, and orient the camera so the sensor is pointing at the illumination source.
4) Place 2 pieces of paper over the sensor opening.
5) Find the maximum exposure duration that just causes the image to saturate. Reducing the duration one increment should result in very few saturated pixels, as indicated by the luminosity histogram. Approximate values are okay.

6) Keeping the exposure duration constant, capture about 10 images, adding another piece of paper to the diffuser stack each time. Continue capturing images until the image is completely black. Alternately, just capture as many images as you have pieces of paper.

Critical Information

Please fill out the following information about your camera. Include your contact info if you wish to be updated with results or if you are interested in participating further.

Camera model: ______________________

Approximate purchase date: ______________________

Has the camera been serviced before?______________________